2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing

iPMVM 2020, November 16–18, 2020, Schloss Dagstuhl, Wadern, Germany (Virtual Conference)

Edited by

Christoph Garth
Jan C. Aurich
Barbara Linke
Ralf Müller
Bahram Ravani
Gunther H. Weber
Benjamin Kirsch

Editors

Christoph Garth
Technische Universität Kaiserslautern, Germany
garth@cs.uni-kl.de

Jan C. Aurich
Technische Universität Kaiserslautern, Germany
jan.aurich@mv.uni-kl.de

Barbara Linke
University of California, Davis, CA, USA
bslinke@ucdavis.edu

Ralf Müller
Technische Universität Kaiserslautern, Germany
ram@rhrk.uni-kl.de

Bahram Ravani
University of California, Davis, CA, USA
bravani@ucdavis.edu

Gunther H. Weber
University of California, Davis, CA, USA
ghweber@lbl.gov

Benjamin Kirsch
Technische Universität Kaiserslautern, Germany
benjamin.kirsch@mv.uni-kl.de

ACM Classification 2012
Applied computing → Engineering; Computing methodologies → Modeling and simulation; Human-centered computing → Visualization

ISBN 978-3-95977-183-2

Published online and open access by
Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at https://www.dagstuhl.de/dagpub/978-3-95977-183-2.

Publication date
April, 2021

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at https://portal.dnb.de.

Digital Object Identifier: 10.4230/OASIcs.iPMVM.2020.0

ISBN 978-3-95977-183-2 **ISSN 1868-8969** **https://www.dagstuhl.de/oasics**

OASics – OpenAccess Series in Informatics

OASics is a series of high-quality conference proceedings across all fields in informatics. OASics volumes are published according to the principle of Open Access, i.e., they are available online and free of charge.

ISSN 1868-8969

https://www.dagstuhl.de/oasics

Contents

Regular Papers

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Preface

The International Research Training Group 2057 "Physical Modeling for Virtual Manufacturing Systems and Processes" funded by the German Research Foundation (DFG) is aimed at enabling the planning of production processes on a new level by incorporating computational and physical models. Computer models are already in use to plan production processes ranging from a single machine to a complete factory; however, these models are lacking a description of the physical properties and processes involved and thus are of limited accuracy and predictive power. In the envisioned new generation of models that include physical aspects, it will be possible to calculate key properties of a production line, such as the quality of the products or the energy consumption of a factory, in advance, and to perform targeted improvements. Within IRTG 2057, the physical interactions of the three levels factory, machine and process are considered. Toward this goal, the research agenda is driven by fundamental problems in both engineering and computers science, as well as by the integration of both. As an international program, IRTG 2057 brings together investigators and students from the three partner universities Technische Universität Kaiserslautern, University of California Davis, and University of California Berkeley.

This volume contains the proceedings of the 2nd International Conference of the IRTG 2057, conducted November 16-18, 2020 in a virtual setting. The topics considered at the conference mirror IRTG 2057's research focus. The 19 contributions contained within this book underwent a two-stage, rigorous review by an international program committee. Submitted papers were accepted for presentation at the conference, and subsequently reviewed again before final acceptance.

We would like to express our immense gratitude to all authors that submitted a paper and the members of the program committee for their diligent work. Finally, we are indebted to the German Research Foundation (DFG, Deutsche Forschungsgemeinschaft) for continued funding and support under contract number 252408385.

February 2021

Christoph Garth, Jan C. Aurich,
Barbara Linke, Ralf Müller, Bahram Ravani
Gunther H. Weber, Benjamin Kirsch

IRTG 2057
Physical Modeling for
Virtual Manufacturing Systems and Processes

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

List of Authors

Jan C. Aurich (1, 2, 3, 7, 11, 15, 17)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Rapheal Bilz (16)
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Georgis Bulun (14)
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Christopher R. Chighizola (11)
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Felix Claus (6)
Computer Graphics & HCI Lab, Computer Science Department, Technische Universität Kaiserslautern, Germany

Ricardo Dutra da Silva (18)
Department of Informatics, Federal University of Technology, Curitiba, PR, Brazil

Kristin M. de Payrebrune (16)
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Christopher R. D'Elia (11)
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Matthias Eifler (14)
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Tim Ettrich (9)
Technische Universität Kaiserslautern, Germany

Rida T. Farouki (5)
Mechanical and Aerospace Engineering Department, University of California Davis, CA, USA

Destiny R. Garcia (5)
Mechanical and Aerospace Engineering Department, University of California Davis, CA, USA

Christoph Garth (4, 10, 19)
Faculty of Computer Science, Technische Universität Kaiserlautern, Germany

Moritz Glatt (15)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Sebastian Greco (3)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Hans Hagen (6, 10)
Computer Graphics & HCI Lab, Computer Science Department, Technische Universität Kaiserslautern, Germany

Bernd Hamann (6, 18)
Department of Computer Science, University of California, Davis, CA 95616, USA

Hans Hasse (17)
Laboratory of Engineering Thermodynamics (LTD), Technische Universität, Kaiserslautern, Germany

Marius Heintz (2)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Michael R. Hill (11)
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Anis Idrizovic (14)
The Institute of Optics, University of Rochester, NY, USA

Julianne E. Jonsson (11)
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Andrej Keksel (14)
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Eberhard Kerscher (17)
Materials Testing (AWP), Technische Universität Kaiserslautern, Germany

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Eric Georg Kinner (19)
Faculty of Computer Science, Technische Universität Kaiserslautern, Germany

Benjamin Kirsch (2, 3, 7, 11, 17)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Katja Klauer (3)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Charlotte Kuhn (9)
Faculty 7, Universität Stuttgart, Germany

Andreas Lange (2)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Heike Leitte (6, 10)
Visual Information Analysis, Computer Science Department, Technische Universität Kaiserslautern, Germany

Viktor Leonhardt (6)
Scientific Visualization Lab, Computer Science Department, Technische Universität Kaiserslautern, Germany

Barbara S. Linke (5, 11)
Mechanical and Aerospace Engineering Department, University of California Davis, CA, USA

Anna-Pia Lohfink (4)
Faculty of Computer Science, Technische Universität Kaiserlautern, Germany

Jonas Lukasczyk (19)
Arizona State University, Tempe, AZ, US

Ross Maciejewski (19)
Arizona State University, Tempe, AZ, US

Daniel Mannherz (16)
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Daniel Müller (7)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Ralf Müller (9, 13)
Applied Mechanics, Technische Universität Kaiserslautern, Germany

Helio Pedrini (18)
Institute of Computing, University of Campinas, SP, Brazil

Mahesh R.G. Prasad (16)
ICAMS, Ruhr-Universität Bochum, Germany

Juan Manuel Rodríguez Prieto (13, 16)
Mechanical Engineering Department, Universidad EAFIT, Medellín, Colombia

Bahram Ravani (8, 15)
Department of Mechanical and Aerospace Engineering, University of California at Davis, CA, USA

David Honegger Rogers (19)
Los Alamos Research Laboratory, NM, US

Patrick Ruediger (10)
Visual Information Analysis Group, Technische Universität Kaiserslautern, Germany

Sebastian Schmitt (17)
Laboratory of Engineering Thermodynamics (LTD), Technische Universität Kaiserslautern, Germany

Christoph Schreiber (9)
Applied Mechanics, Technische Universität Kaiserslautern, Germany

Jörg Seewig (14)
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Ardalan R. Sofi (8)
Department of Mechanical and Aerospace Engineering, University of California at Davis, CA, USA

Praveen Sridhar (16)
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Simon Stephan (17)
Laboratory of Engineering Thermodynamics (LTD), Technische Universität Kaiserslautern, Germany

Herbert M. Urbassek (12, 17)
Physics Department and Research Center OPTIMAS, Technische Universität Kaiserslautern, Germany

Vardan Hoviki Vardanyan (12)
Physics Department and Research Center OPTIMAS, Technische Universität Kaiserslautern, Germany

Daniel Weber (11)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Xialong Ye (13)
Applied Mechanics, Technische Universität Kaiserslautern, Germany

Li Yi (1)
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Development and Validation of Energy Simulation for Additive Manufacturing

Li Yi[1]
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Jan C. Aurich
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Abstract

Additive manufacturing (AM) is a promising manufacturing technology towards cleaner production systems. Nevertheless, recent studies state that environmental benefits of AM are case-specific and need to be evaluated and confirmed in the design phase. To enable the energy performance evaluation in the design phase, developing convenient tools for energy prediction of AM has been an important research task. Aiming at this problem, this paper presents the research for energy modeling, simulation implementation, and experimental validation of an energy simulation tool of two AM processes: Selective laser melting (SLM) and Fused deposition modeling (FDM). The developed simulation tool can be conveniently used for energy consumption quantification and evaluation during the product and process design for AM.

2012 ACM Subject Classification Applied computing → Computer-aided design

Keywords and phrases Additive manufacturing, fused deposition modeling, selective laser melting, energy simulation, eco-design for AM

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.1

Funding This research is funded by the Deutsche Forschungsgemeinschaft (DFG German Research Foundation) - 252408385 - IRTG 2057.
Li Yi: DFG IRTG 2057 252408385.
Jan C. Aurich: DFG IRTG 2057 252408385.

1 Introduction

Additive manufacturing (AM) is a promising technology to improve sustainable performance of production systems [16]. AM enables the design and manufacturing of novel geometrical features like lattices and hollow bodies which are not possible or only with high expenses to produce with conventional manufacturing technologies [27]. The enhanced design freedom implies better functionalities and more environment-friendly performance of products [15]. Besides, AM requires no tools, dies, or lubricants, and therefore, the absence of peripheral accessories and materials lead to savings of resources that would be used to manufacture, maintain, transport and operate them [32]. In AM, components can be produced close to their final desired shapes, leading to less material waste and energy consumption.

Nevertheless, recent studies state that the environmental benefits of AM should be regarded more critically, especially in terms of energy performance [30]. For example, Kellens et al. state that specific energy consumptions (SEC) of AM processes can be one or two orders of magnitude higher than conventional cutting or casting processes [24], and Gutowski et al. observe that adiabatic efficiency of laser-based AM machines for melting steel powder

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 1; pp. 1:1–1:17
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

mainly ranges from 9 % to 23 %, and for aluminum powder ranges from 3.6 % to 7 % [14]. The SEC for producing different metal powders ranges from 7.02 MJ/kg to 23.8 MJ/kg depending on materials and processes [24]. Moreover, Baumers et al. and Peng et al. state that the environmental benefits of AM should be evaluated and confirmed in the design phase before AM is implemented [2, 30]. Therefore, energy consumption prediction and evaluation of AM in the design phase have been emerging research topics. Current methods for energy measurements of AM mainly rely on experiments (e.g., [25, 10, 9]), which are not appropriate for the design phase because it is time- and cost-spending, since process time of an AM process can be up to days or weeks depending on the AM systems and process parameters. Aiming at this problem, this paper presents the development and validation of an energy simulation tool of AM, which enables a fast energy prediction for a given product design and AM system in the design phase [39, 40].

2 Background

2.1 Additive manufacturing (AM)

2.1.1 Definition and terms of AM

The standard ISO/ASTM 52900 defines the term "Additive Manufacturing" as " (...) process of joining materials to make parts from 3D model data, usually layer upon layer, as opposed to subtractive manufacturing and formative manufacturing methodologies (. . .)" [19]. The German standard VDI 3405 proposes a similar definition that an AM process is regarded as "(..) manufacturing process in which the workpiece is built up in successive layers or units (. . .)" [36]. In addition, different synonyms like 3D printing, layered manufacturing, generative manufacturing, free-form fabrication can be found in literature. The most nominal one is "3D printing", which indicates that AM processes are "(..) 3D analog to ubiquitous 2D printers (. . .)" [5]. In 2009, the ASTM F42 Technical Committee on AM hold a meeting in West Conshohocken, Pennsylvania, in which the term "Additive Manufacturing" was formally selected as the name of processes for joining materials to create parts [5]. Afterwards, the term "Additive Manufacturing" was embraced by ISO Technical Committee TC 261, which is responsible for developing a series of standards for AM processes. Today, the term "Additive Manufacturing" has been adopted internationally by research communities.

2.1.2 Difference between AM and conventional manufacturing (CM)

Generally, a manufacturing process is described as a combination of operations for the production of geometrically defined solid bodies (German: " (..) Verfahren zur Herstellung von geometrisch bestimmten festen Körppern (. . .)") [8]. Considering the shaping mechanisms of materials, manufacturing processes can be distinguished between the following three types, as depicted in Figure 1 [19]:

- **Formative shaping**, in which the desired geometry is created by applying pressure to a body of raw material, e.g., casting and bending;
- **Subtractive shaping**, in which the desired geometry is created by selective removal of materials, e.g., drilling, turing, and milling;
- **Additive shaping**, in which the desired geometry is made by successive addition of material

The main difference between them is the volume change of a workpiece [11]. In formative shaping, by assuming a constant material density, the volume of a workpiece before and after the shaping is constant, see V_1 and V_0 depicted in Figure 1. In subtractive manufacturing,

V_1 is smaller than V_0 because of the material removal, while only in additive shaping, V_1 is larger than V_0 because the material is applied layer by layer. Therefore, the changing mechanism of volume during the material shaping is a fundamental criterion to distinguish AM processes from other manufacturing processes.

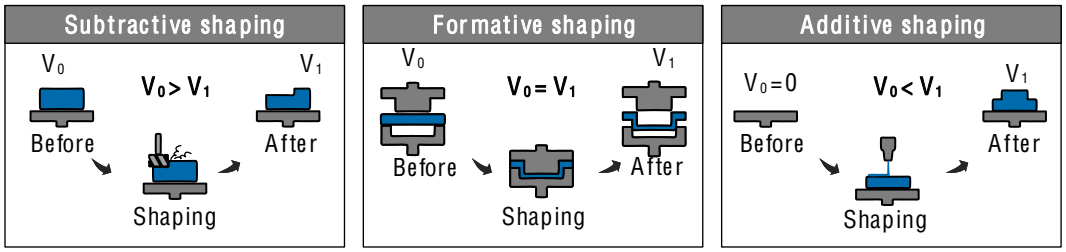


Figure 1 Distinguish between AM and conventional manufacturing (according to [11]).

2.1.3 AM process categories

The standard ISO 17296 has proposed seven major AM process categories [18]. Each category is characterized by its own processable feedstocks, system characteristics, and suitable application areas, as described in the following and depicted in Figure 2 [38, 18, 31]:

- **Vat photopolymerization** uses a light source (e.g., UV radiation) to harden photo-curable liquid or ceramic paste with mixed photo-curable materials, and it is the first commercialized AM process. Vat photopolymerization can be used to produce prototypes, patterns, or presentation models due to its high accuracy and fine surface finishing.
- **Material jetting** uses a spray nozzle to deposit droplets of build materials on a platform. The most common feedstock is liquid or semi-liquid made by single material or composites. By defining the composition of different materials, material jetting process is capable of processing components with unique electrical, optical or other physical properties [38, 13].
- **Binder jetting** dispenses bonding agent to a selective area of a powder bed and can be used to produce plastics, metals, or ceramics, as long as the material is in powder form. By using multiple nozzles, material jetting can achieve a higher build rate and can be used to produce prototypes or functional parts. For printed metal or ceramic parts, a post-sintering process is required to fully densify the part.
- **Material extrusion** squeezes filamentary materials or paste (e.g. plastics or structural ceramics) on a platform to create a part. Due to its convenient usability and low-priced machines and materials, material extrusion has been widely adopted for rapid prototyping. Today, material extrusion can be used to produce metal parts if metal particles are filled with a binder matrix into filaments.
- **Powder bed fusion** uses thermal energy like laser beam or electron beam to fuse powders in selective region of a powder bed. Today, powder bed fusion is one of the most important process categories for producing metal parts delivering complex functionalities, i.e., tools or end use parts.
- **Directed energy deposition** uses focused thermal energy to create a local molten pool, in which the material powder or wire is continuously feed and melted. Directed energy deposition is another promising process category for producing metal parts and can be integrated into conventional milling machine tools to enable a hybride additive-subtractive manufacturing.

- **Sheet lamination** bonds sheets of material to form an object. The range of the processable materials is very wide, as sheets can be metal, plastic, ceramic, paper, wood or composite. Sheet lamination is suitable for producing models, patterns, or prototypes that have less requirement on geometrical complexity and accuracy.

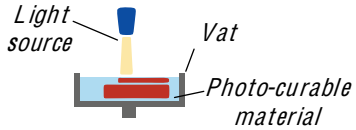
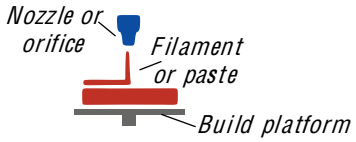
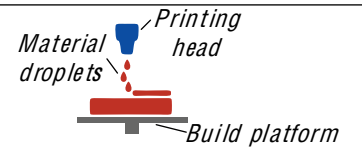
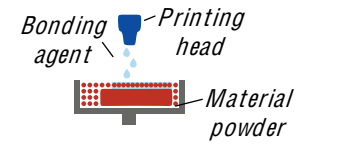
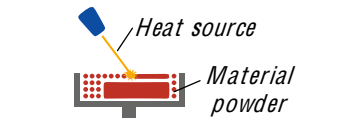
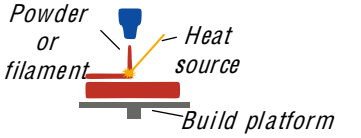
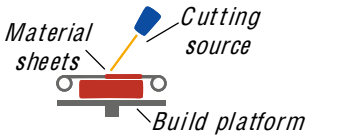
Process category	Schema	Material	Application
Vat photopolymerization		• Polymer • Ceramic	• Prototyping • Functional testing • Tooling patterns • Detailed parts • Presentation model
Material extrusion		• Polymer • Metal • Ceramic • Composite	• Prototyping • Functional testing • Tooling patterns • Personal use
Material jetting		• Polymer • Metal • Ceramic • Composite	• Concept model • Limited functional testing • Colored design models
Binder jetting		• Polymer • Metal • Ceramic	• Prototyping • End use parts • Casting/forming tools
Powder bed fusion		• Polymer • Metal • Ceramic	• End use parts • Functional testing • Rapid tooling • High-temperature applications
Directed energy deposition		• Metal	• End use parts • Functional testing • Rapid repair/overhaul • High-temperature applications
Sheet lamination		• Polymer • Metal • Ceramic • Composite • Paper	• Form testing • Tooling patterns • Less detailed parts

Figure 2 Classification of AM processes (own illustration according to [18]).

2.2 Energy performance of AM

2.2.1 Debates on the energy performance of AM

Energy is a physical property that can be transferred between objects and expressed by different forms, while energy performance describes the energy consumption, energy efficiency, and energy usage of a system or process [17]. In the current research, there are two different views on the energy performance of AM. One group of researchers believes that AM has the

potential to reduce energy consumption in manufacturing industries, while the other group believes that AM's potential may be very limited and difficult to realize. On the positive side, the following perspectives are observed:

- **Lightweighting of products**: AM enables the lightweighting using topology optimization, lattice structures, hole body, lightweighting materials etc. On the one hand, lightweighting leads to less material usage for manufacturing a product, and on the other hand, lightweighted products can be beneficial for the use phase. For example, aircraft with less weight requires less fuel consumption [16].
- **Less material waste**: For powder-based AM processes like selective laser melting (SLM), the powders that are not used in a build task can be recycled, screened, and reused [32]. The scraps with AM are less than conventional manufacturing, and they can only be found in limited places, e.g., support structure.
- **Absence of product-specific tools or peripheral substances**: AM requires no dies, cutting tools, fluids, or other auxiliary substances or devices. Therefore, their production, transportation, storage, operation, maintenance, and disposal are not required, which indicates that energy usages related to these steps are saved [27].
- **Shortening of supply chains**: AM enables assembly integration in which multiple parts are integrated to single complex parts. Therefore, the relevant manufacturing steps and secondary processes (e.g., storage and transport) are reduced, leading to a shorter and more flexible supply chain. By 2025, primary energy and CO_2-emission intensities of industrial manufacturing supply chains can be reduced by up to 5% through AM [12].
- **Rapid repair and remanufacturing**: At the use phase, products can be repaired by AM, and hence, the lifecycles of products are extended. At the end-of-life phase, products can be remanufactured and reused. Finally, Extension of the lifecycle of an existing product enables the saving of energy and materials that would be used for producing a new one [26]

In addition to these positive views, the following concerns about potentials of AM to improve the energy performance still exist:

- **Production of feedstock**: Different AM processes require different shapes of feedstock, while conventional manufacturing is not very sensitive to the shape of feedstock [24]. For metal powder-based AM processes like binder jetting and powder bed fusion, metal powers should be produced by water or gas atomization that requires a significant amount of energy [6].
- **Need of post-processing**: While cutting tools or fluid are not needed during the in-process of AM, they are still used during the post subtractive processing, in order to improve the surface quality and geometrical accuracy of AM parts [35]. Thus, the tools and auxiliary substances that are claimed to be absent with AM are still not avoidable. Consequently, the energy usage that should be saved due to their absence is still there.
- **Lower efficiency and high SEC** In metal AM processes, the heat exchange between heat source and metals cause high energy waste due to radiation, reflection, conduction, and convection. The adiabatic efficiency of laser AM process (ratio between actual build rate with heat loss and theoretical maximal build rate without heat loss) ranges from 3.6% to 7% for aluminum powders, and from 9% to 23% for steel powders [14]. Moreover, SEC values of AM processes can be 1 to 2 orders of magnitude higher than conventional machining and molding processes [24].
- **Long process time** Build time of different AM processes ranges from hours to days depending on machines and process parameter sets[1]. Longer process time implies more electricity consumption and a higher risk of failure. Once an error occurs during the process, the build task needs to be repeated, leading to more material and energy expenses.

2.2.2 Current research questions related to energy issues of AM

In accordance with the debates on the energy performance of AM, the following research questions are essential at the current research background:

- **Quantitative energy performance assessment of AM processes**: The energy performance of AM can be influenced by different factors such as part size, build orientation, batch size, process parameters, machine configuration, etc [24, 2]. Thus, the quantification and evaluation of different AM processes and systems with varying influence factors are important research tasks.
- **Comparison of AM with conventional manufacturing in multiple life stages**: In different life phases, AM may have advantages or even disadvantages compared to conventional manufacturing, e.g., production, usage, service, disposal [2]. Therefore, energy performance of AM versus conventional manufacturing should be compared throughout the lifecycle to avoid creating a simplistic picture [2].
- **Raw materials and feedstock**: Current literature has only limited studies addressing technical and environmental issues of production of feedstock for AM, e.g., [9, 29]. The energy consumption and primary resource usage for material extraction and feedstock production of AM should be quantified and evaluated.
- **Eco-design**: Since energy performance of AM may be a critical problem, how to improve the energy performance of AM has been another important research task [2]. A promising approach is eco-design in which environmental aspects are considered in design phase, and future works should pay more attention on this issue [2, 30].

2.2.3 Research questions for this research

In accordance with the current research background, this research specifically aims at the development and validation of an energy simulation for AM. Choosing this research question encompasses the following three motivations:

- **Systematical exploration and understanding of the energy consuming behavior of AM**: AM systems consume electricity and transfer it into movement of platforms and heat to melt materials. Thus, AM processes involve different forms of energy, and a holistic description and understanding of them is the prerequisite to optimize the energy performance of AM.
- **Convenient and accurate energy prediction in the design phase**: As mentioned before, eco-design for AM requires the quantification and evaluation of energy performance of AM in the design phase. However, current methods to quantify energy consumption of AM systems are experiments that are time- and cost-spending. Therefore, proposing a reliable simulation tool will significantly contribute to supporting eco-design for AM.
- **Lack of predictive models of energy consumption for AM**: In the current literature, only very few studies mentioned the development of energy predictive models for AM, e.g., Baumers et al. proposed empirical methods (black box approach) for energy prediction of SLM process [3]. Our research adopts a physical method (white box approach) in which the energy flow of AM is modeled by physical variables, leading to a methodological contribution to current literature.

3 Methodology

3.1 Reference AM processes and machines

In this research, two AM systems are selected as the referential research objects. The first one is Concept Laser Mlab, which is categorized in SLM process [7]. Concept Laser Mlab is equipped with a laser device for producing metal parts, and the feedstock includes different metal powders such as aluminium, steel, and titanium alloy. The maximal size of a build platform is 90×90 mm^2, and scan speed of the laser beam can be up to 7 m/s. During a build process, inert gas is required to avoid oxydation. For different materials, different inert gas is needed, e.g., argon for aluminium, and nitrogen for steel. The second AM system is Ultimaker 3 that belongs to FDM process and is capable of processing thermal plastics such as PLA, ABS, PVA, and TPU [33]. Ultimaker 3 is equipped with two extruders that melt and extrude build material and support material, respectively. After the part is printed, the support structure needs to be revomed. Figure 3 shows photos of the two AM systems including their auxiliary devices and examples of manufactured parts.

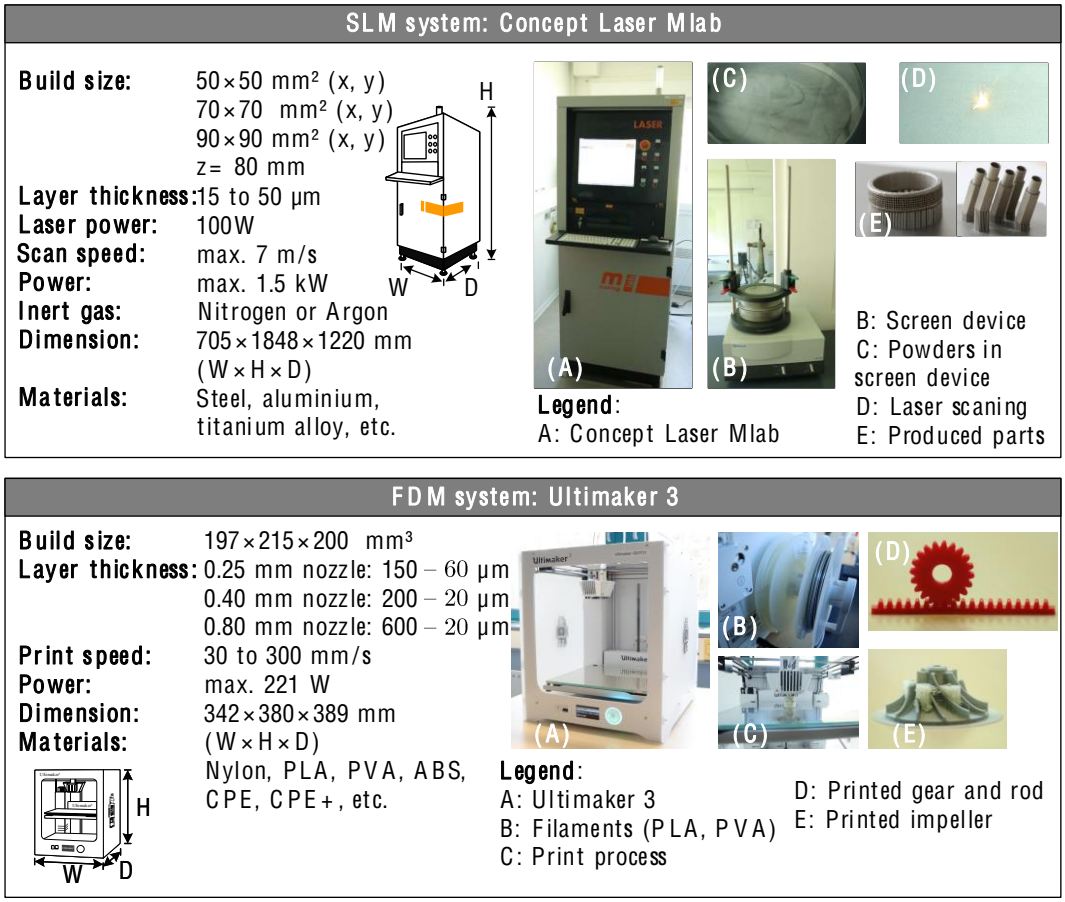


Figure 3 Reference AM systems for this research.

3.2 System boundary and approach

3.2.1 Definition of system boundary

Manufacturing can be discussed between different levels, as depicted in Figure 4 [37]. In this research, the research focus is limited to the workstation/machine level, neglecting other issues like production feedstock or post-processing. According to standard ISO 14955-1, the system boundary for energy evaluation of a machine tool should include the machine tool itself and its auxiliary devices [20]. Thus, this research adopts this standard and considers an AM system as a system containing at least one AM machine and its peripherals such like screen device and vacuum cleaner.

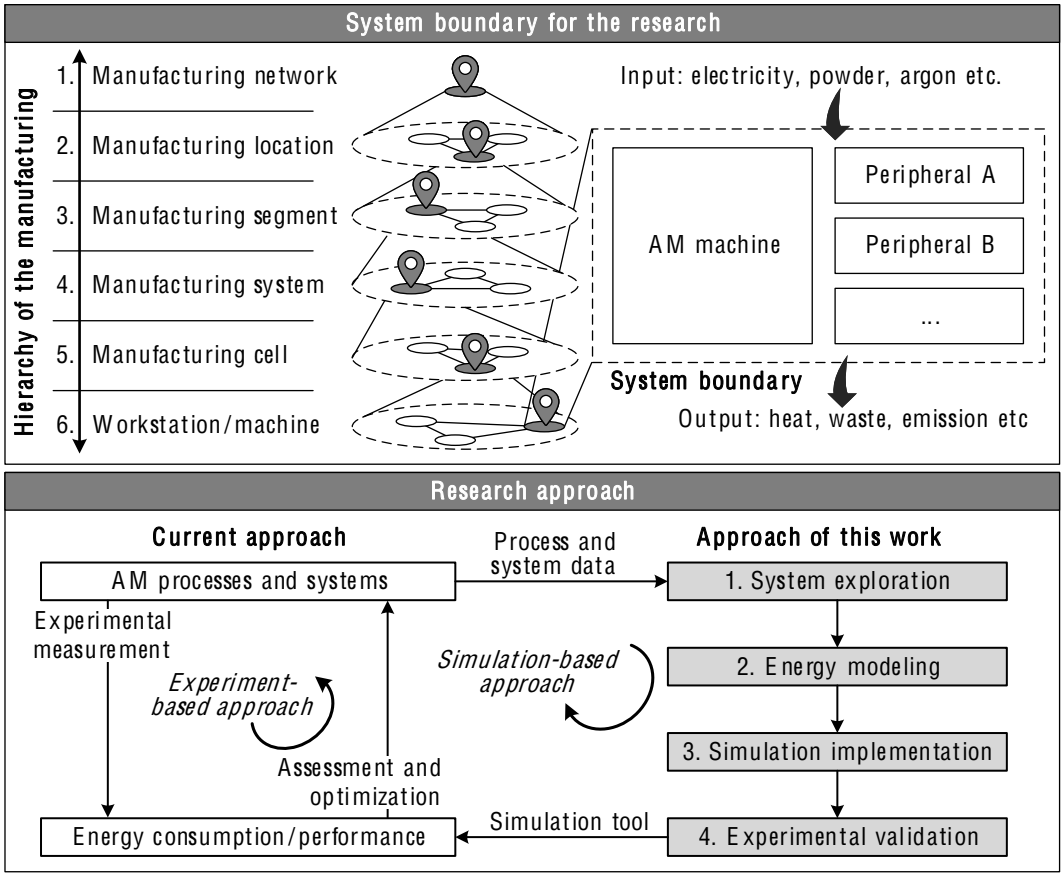


Figure 4 Research approach and system boundary.

3.2.2 Research approach

Compared to conventional experiment-based approach for measuring and analyzing the energy consumption of AM, our research approach for the energy simulation development and validation encompasses the following four phases, as illustrated in Figure 4:

- **System exploration**: In this phase, the structure of the AM-machine and the functions of system components related to power consumption or exchange are analyzed. Furthermore, the system components that should be considered in the energy modeling are defined, and their information are collected from internal or external data sources, e.g., data sheets and circuit diagrams.

- **Energy modeling** In the second phase, the energy flows between system components are modeled using bond graph because bond graph enables the modeling of power flows between multiple physical domains based on a unified terminology. [23]. First, the related energy domains of the energy flow are analyzed and the power variables and multiports are defined. Afterwards, the bond graph of the AM system is created. For more information about bond graph methodology refer to [23, 4].
- **Simulation implementation**: In the third phase, based on the bond graph, a simulation tool is developed based on the MATLAB® platform. First, based on the bond graph, the equivalent simulation models of system components are created using Simulink. Afterwards, by running the Simulink models, a power database is generated. By programming a GUI using the function module *AppDesigner* of MATLAB®, an energy simulation tool is developed. The simulation approach developed in this paper is called Numerical Control (NC) code (also called G-code) and database-driven approach, which is explained in the appendix.
- **Experimental validation**: In the last step, experiments are carried out to verify the simulation accuracy of the simulation tool. The power meter is YOKOGAWA WT 1806 [41], and wiring of the measurement adopts standard ISO 14955-2 [21]. For capturing the power data, the sampling time is set as 1 s.

4 Result and discussion

4.1 Energy simulation tool

4.1.1 Data structure and simulation logic

The data structure of the simulation is illustrated in a functional flow block diagram, as depicted in Figure 5.

The first function is the import of .stl-file and .lsr-file (or .gcode-file), from which the facet data of a part and the layer data of a process are extracted, respectively. Afterwards, while the facet data are used to visualize the geometry, the layer data are used to facilitate the energy simulation. After the simulation, the power curves of the AM machine and peripherals are visualized , and the simulation result including power curves and data tables can be exported.

The core function of the simulation tool is *energy simulation* containing eight subfunctions, as shown in Figure 5. Followed by the third function *extract layer data*, time parameters are converted to the time array with a sampling time of 1 s. To mention is that the sampling time cannot be too short. For conventional manufacturing like milling or drilling, the sampling time for obtaining power values can be set to 0.1 s because the process lasts only minutes or tens of minutes [22]. However, for AM processes, the build time can be up to hours or days, and shorter sampling time implies more frequent sampling, leading to a massive volume of data. To avoid this, the sampling time to obtain power values is defined as 1 s. After generating the time array, power values for each system component and point in time are generated from on the database. Finally, after the trapezoidal integration of power and time arrays, the energy values are calculated.

In addition to the NC code-driven approach, this research adopts the use of a power database. In a previous idea, Simulink models are directly integrated to GUI in which the power data of system components are simulated after the time parameters are read. However, the simulation of all these Simulink models is extremely time-spending. Adding a power database, in which the power data of Simulink models are already stored, can significantly reduce the simulation time. The sacrifice for this decision is a reduced flexibility for AM

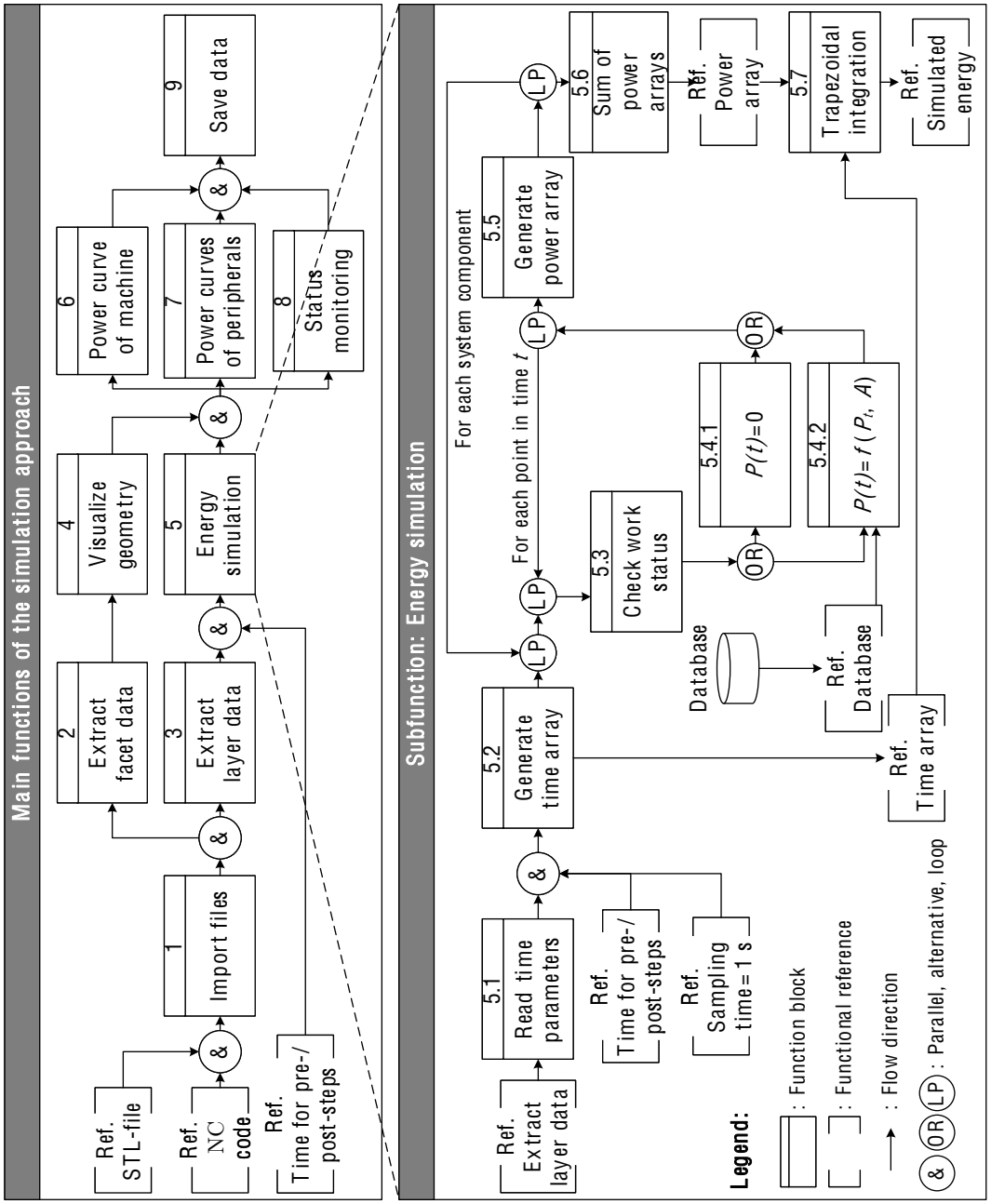


Figure 5 Functional logic of the simulation tool.

system modification (e.g., add a second laser device or change the platform motor). If Simulink models are directly connected with GUI, users need to carry out two steps for system modification: first, change Simulink models, and second, run the energy simulation in GUI. However, if the GUI is connected with a power database, users have to carry out three steps: first, modify the Simulink models, and second, run the Simulink models and update the power database, and third, run the energy simulation in GUI. Nevertheless, if user do not change the AM system configuration, but only change process parameters or product design, the power database-driven simulation can reduce the simulation time from hours to seconds.

4.1.2 Graphical user interface (GUI)

The GUI of the developed simulation software is depicted in Figure 6. The simulation approach is called NC code-driven approach, in which NC code of a process design should be generated at first and then imported to the GUI to start a simulation. Figure 6 shows examples of the NC codes, in which movement command, location information, and time parameters are contained. In this research, *.lsr* and *.gcode* file formats are used for the simulation of Concept Laser Mlab and Ultimaker 3, respectively, and they can be exported by the software Netfabb and Ultimaker Cura, respectively [28, 34].

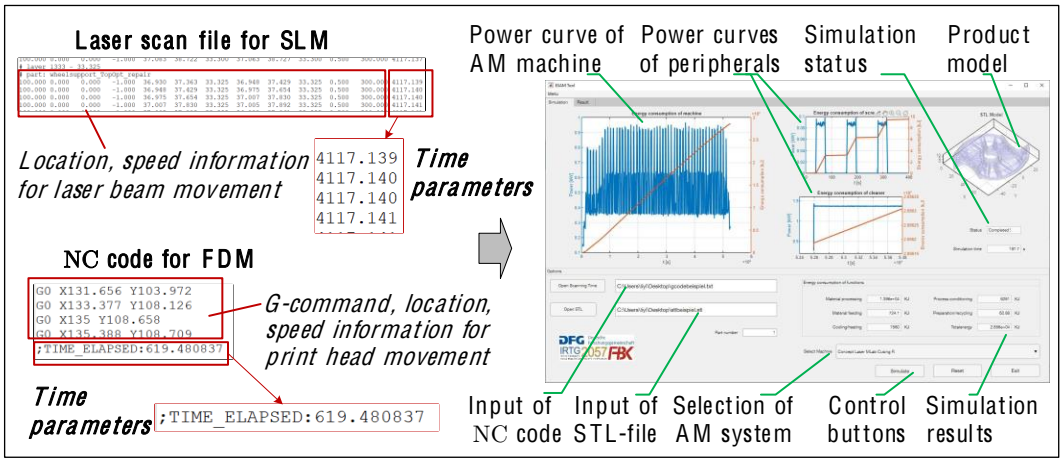


Figure 6 GUI of the simulation software.

4.2 Analysis of the simulation accuracy

4.2.1 Accuracies for SLM

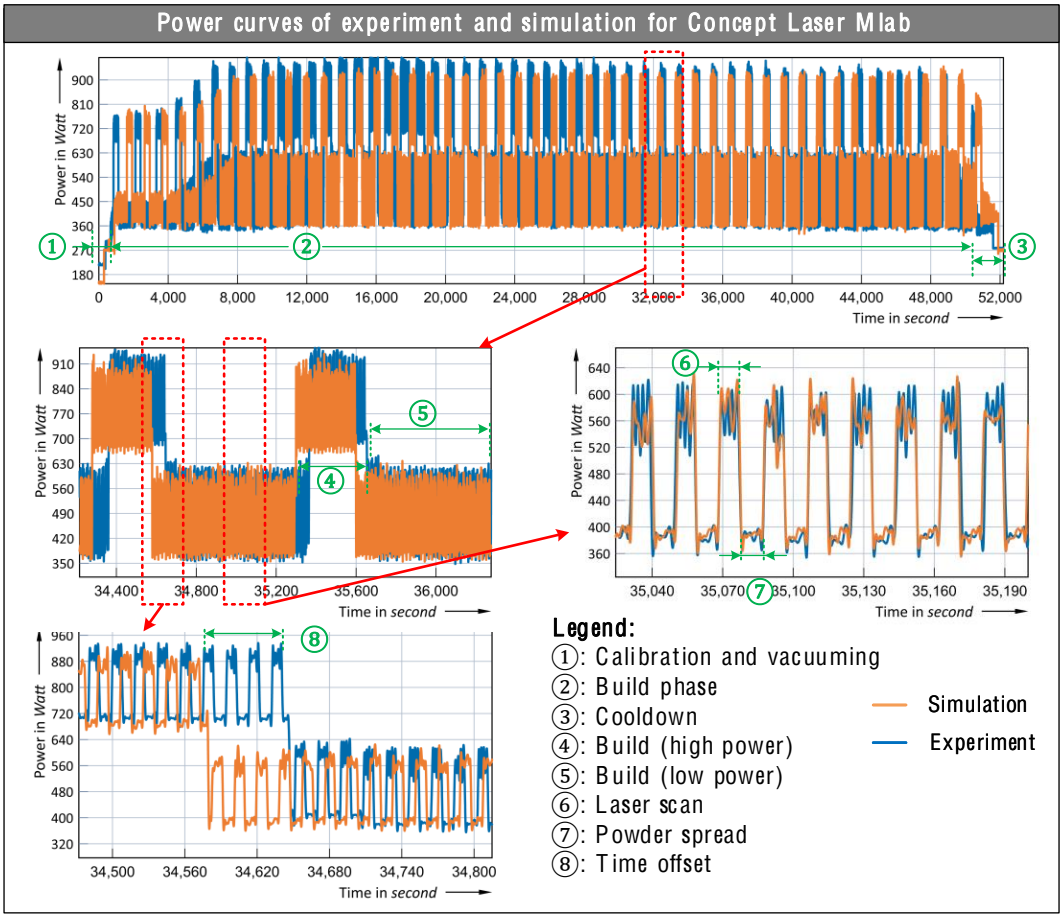
The power curves of the simulation and experiments for Concept Laser Mlab are depicted in Figure 7 in which the power curve can be characterized by three stages. While the first stage indicates the calibration and vacuuming of the build chamber (①), the second and third stages describe the build of part ((②)) and the cooldown ((③)), respectively. The second stage can be further divided into two build modes, as indicated by ((④)) and ((⑤)) in Figure 7. The cyclic operation of the compressor in cooling device leads to a cyclical power increase by approximately 300 W. By zooming the power curve, another periodical power increase is observed, as indicated by ((⑥)) and ((⑦)), implying the laser scanning and powder spreading, respectively. The laser scanning leads to a power increase by approximately 210 W.

Although the simulated power curve and the actual power curve generally fit, there are still deviations. One of them is the time offset, as marked by ((⑧)). The reason is that time parameters in NC code files used for simulations vary from the actual time parameters. Thus, the timelines of simulations and experiments are not perfectly matching up.

The table in Figure 7 summarizes the result of simulations and experiments, including the calculated evaluation indicators. The simulation accuracy ACC_E is defined by the following equation, in which E_{sim} and E_{ext} represent the energy consumption of simulation and experiment respectively:

$$ACC_E = \frac{\Delta E}{E_{exp}} = \frac{|E_{sim} - E_{exp}|}{E_{exp}} \quad (1)$$

In four experiments, mean ACC_E is 96.43%. Since it is higher than 95%, it can be concluded that the simulation accuracy is verified and the simulation tool can be used for energy performance quantification and evaluation of Concept Laser Malb.



No.	n_{layer}	t_{build} (min)	E_{exp} (kWh)	E_{sim} (kWh)	ΔE (kWh)	ACC_E (%)
1	2546	874.48	8.03	7.91	0.12	98.51
2	433	181.15	1.41	1.48	0.07	95.04
3	605	153.00	1.10	1.14	0.04	96.36
4	1113	403.40	2.39	2.29	0.10	95.82

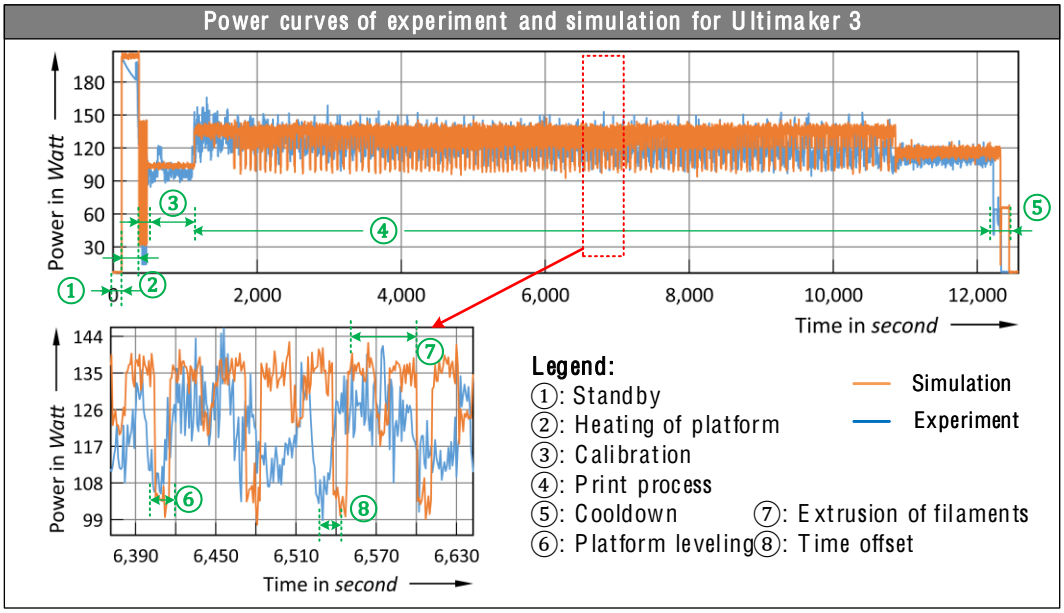
Figure 7 Power curves and experimental and simulated result for Concept Laser Mlab.

4.2.2 Accuracy for FDM

In Figure 8, the simulated and experimental power curves of Ultimaker 3 are depicted, in which five stages are characterized. In the first stage standby, as indicated by ①, the power consumption is approximately 5 W, while during the second stage (②), the platform is heated to 60 °C, and the power consumption is increased to approximately 210 W. Afterwards, in the third and fourth stages, indicated by ③ and ④ respectively, the calibration of the print head and the print process are executed. The power consumption of print process is approximately 130 W. Finally, in the last stage cooldown (⑤), the power consumption falls back to 5 W.

The print process depicts a cyclical behavior, which is caused by the leveling of platform and extrusion of filaments, indicated by ⑥ and ⑦ respectively. Moreover, the time offset is also observed, in which the timeline from the NC code file varies from the real timeline, as seen as ⑧ in Figure 8.

In the experimental verification, eight experiments are performed, with the mean ACC_E of 94.7 %. Main reason for this deviation is the time offset that can be easily impacted by environmental factors. For example, if Ultimaker 3 is turned on in a morning with lower room temperature, it requires more time to heat up and causes higher deviation in time parameters. The solution to eliminate or to reduce these unpredictable influence can be the standardization of the application scenario and workflow for operating Ultimaker 3.



No.	t_{build} (min)	E_{exp} (kWh)	E_{sim} (kWh)	ΔE (kWh)	ACC_E (%)
1	203.18	0.41	0.42	0.01	97.6
2	207.53	0.41	0.43	0.02	95.1
3	264.57	0.52	0.54	0.02	96.2
4	268.60	0.55	0.56	0.01	98.4
5	81.35	0.16	0.15	0.01	94.9
6	83.10	0.16	0.15	0.01	93.3
7	105.12	0.21	0.19	0.02	90.5
8	105.47	0.21	0.19	0.02	91.4

Figure 8 Power curves and experimental and simulated result for Ultimaker 3.

5 Conclusion and outlook

This paper introduces the development and validation of a simulation tool for energy consumption prediction of two representative AM systems. Based on the result of research, the following conclusions can be made:

- Bond graph is a suitable tool for describing the power exchange between system components of AM systems.

- NC code-driven simulation approach enables a high simulation accuracy. In experimental validation, the mean accuracy for both AM systems are approximately 95 %, and the highest accuracy can be up to approximately 98 %.
- Use of power database can significantly reduce the simulation time.
- Time offset is the main reason for causing simulation deviation.

Based on this research, future works should focus on the adaption of this method to other AM processes and systems, as well as implementation of the simulation tool into eco-design for AM methods.

References

1 Mohsen Attaran. The rise of 3-d printing: The advantages of additive manufacturing over traditional manufacturing. *Business Horizons*, 60(5):677–688, 2017. doi:10.1016/j.bushor.2017.05.011.

2 Martin Baumers, Joost R. Duflou, William Flanagan, Timothy G. Gutowski, Karel Kellens, and Reid Lifset. Charting the environmental dimensions of additive manufacturing and 3d printing. *Journal of industrial ecology*, 21(S1):9–14, 2017. doi:10.1111/jiec.12668.

3 Martin Baumers, Chris Tuck, Ricky Wildman, Ian Ashcroft, Emma Rosamond, and Richard Hague. Transparency built-in. energy consumption and cost estimation for additive manufacturing. *Journal of industrial ecology*, 17(3):418–431, 2013. doi:10.1111/j.1530-9290.2012.00512.x.

4 Wolfgang Borutzky. *Bond Graph Methodology*. Springer London, London, 2010. doi:10.1007/978-1-84882-882-7.

5 David L. Bourell, Ming C. Leu, and David W. Rosen. Roadmap for additive manufacturing: Identifying the future of freeform processing. URL: https://wohlersassociates.com/roadmap2009A.pdf.

6 Danfang Chen, Steffen Heyer, Suphunnika Ibbotson, Konstantinos Salonitis, Jón GarÐar Steingrímsson, and Sebastian Thiede. Direct digital manufacturing: Definition, evolution, and sustainability implications. *Journal of cleaner production*, 107:615–625, 2015. doi:10.1016/j.jclepro.2015.05.009.

7 Concept Laser. Mlab cusing r. https://www.concept-laser.de/en/products/machines.html, 2019.

8 DIN. Fertigungsverfahren –begriffe, einteilung (manufacturing processes – terms and definitions, division), 2003.

9 Jeremy Faludi, Martin Baumers, Ian Maskery, and Richard Hague. Environmental impacts of selective laser melting: Do printer, powder, or power dominate? *Journal of industrial ecology*, 21(S1):S144–S156, 2017. doi:10.1111/jiec.12528.

10 Claes Fredriksson. Sustainability of metal powder additive manufacturing. *Procedia manufacturing*, 33:139–144, 2019. doi:10.1016/j.promfg.2019.04.018.

11 Andreas Gebhardt. *Additive Fertigungsverfahren: Additive Manufacturing und 3D-Drucken für Prototyping - Tooling - Produktion*. Hanser, München, 5., neu bearbeitete und erweiterte auflage edition, 2016. URL: http://gso.gbv.de/DB=2.1/PPNSET?PPN=870133330.

12 Malte Gebler, Anton J.M. Schoot Uiterkamp, and Cindy Visser. A global sustainability perspective on 3d printing technologies. *Energy Policy*, 74:158–167, 2014. doi:10.1016/j.enpol.2014.08.033.

13 Ian Gibson, David Rosen, and Brent Stucker. *Additive Manufacturing Technologies*. Springer New York, New York, NY, 2015. doi:10.1007/978-1-4939-2113-3.

14 Timothy Gutowski, Sheng Jiang, Daniel Cooper, Gero Corman, Michael Hausmann, Jan-Anders Manson, Timo Schudeleit, Konrad Wegener, Matias Sabelle, Jorge Ramos-Grez, and Dusan P. Sekulic. Note on the rate and energy efficiency limits for additive manufacturing. *Journal of industrial ecology*, 21(S1):S69–S79, 2017. doi:10.1111/jiec.12664.

15 Timothy J. Horn and Ola L. A. Harrysson. Overview of current additive manufacturing technologies and selected applications. *Science Progress*, 95(3):255–282, 2012. doi:10.3184/003685012X13420984463047.

16 Runze Huang, Matthew Riddle, Diane Graziano, Joshua Warren, Sujit Das, Sachin Nimbalkar, Joe Cresko, and Eric Masanet. Energy and emissions saving potential of additive manufacturing: The case of lightweight aircraft components. *Journal of cleaner production*, 135:1559–1570, 2016. doi:10.1016/j.jclepro.2015.04.109.

17 ISO. Energy management systems - measuring energy performance using energy baselines and energy performance indicators (enpi) - general principles and guidance, 2014. URL: https://www.iso.org/standard/51869.html.

18 ISO. Iso 17296-2:2015. additive manufacturing — general principles — part 2: Overview of process categories and feedstock, 2015.

19 ISO. *ISO/ASTM 52900:2015 (ASTM F2792) Additive manufacturing – General principles – Terminology*. ISO, Geneva, 2015.

20 ISO. Iso 14955-1:2017 machine tools - environmental evaluation of machine tools - part 1: Design methodology for energy-efficient machine tools, 2017. URL: https://www.iso.org/standard/70035.html.

21 ISO. Iso 14955-2:2018 machine tools - environmental evaluation of machine tools - part 2: Methods for measuring energy supplied to machine tools and machine tool components, 2018. URL: https://www.iso.org/standard/66354.html.

22 S. Kara and W. Li. Unit process energy consumption models for material removal processes. *CIRP annals ... manufacturing technology*, 60(1):37–40, 2011. doi:10.1016/j.cirp.2011.03.018.

23 Dean C. Karnopp, Donald L. Margolis, and Ronald C. Rosenberg. *System dynamics: Modeling and simulation of mechatronic systems*. Wiley, Hoboken, NJ, 5th ed. edition, 2012.

24 Karel Kellens, Raya Mertens, Dimos Paraskevas, Wim Dewulf, and Joost R. Duflou. Environmental impact of additive manufacturing processes: Does am contribute to a more sustainable way of part manufacturing? *Procedia CIRP*, 61:582–587, 2017. doi:10.1016/j.procir.2016.11.153.

25 Karel Kellens, Renaldi Renaldi, Wim Dewulf, Jean-pierre Kruth, and Joost R. Duflou. Environmental impact modeling of selective laser sintering processes. *Rapid prototyping journal*, 20(6):459–470, 2014. doi:10.1108/RPJ-02-2013-0018.

26 Maija Leino, Joonas Pekkarinen, and Risto Soukka. The role of laser additive manufacturing methods of metals in repair, refurbishment and remanufacturing – enabling circular economy. *Physics Procedia*, 83:752–760, 2016. doi:10.1016/j.phpro.2016.08.077.

27 W. R. Morrow, H. Qi, I. Kim, J. Mazumder, and S. J. Skerlos. Environmental aspects of laser-based and conventional tool and die manufacturing. *Journal of cleaner production*, 15(10):932–943, 2007. doi:10.1016/j.jclepro.2005.11.030.

28 Netfabb. Connected software for additive manufacturing, design, and simulation, 2019. URL: https://www.autodesk.com/products/netfabb/overview.

29 Henri Paris, Hossein Mokhtarian, Eric Coatanéa, Matthieu Museau, and Inigo Flores Ituarte. Comparative environmental impacts of additive and subtractive manufacturing technologies. *CIRP Annals*, 65(1):29–32, 2016. doi:10.1016/j.cirp.2016.04.036.

30 Tao Peng, Karel Kellens, Renzhong Tang, Chao Chen, and Gang Chen. Sustainability of additive manufacturing: An overview on its energy demand and environmental impact. *Additive manufacturing*, 21:694–704, 2018. doi:10.1016/j.addma.2018.04.022.

31 Onno Ponfoort. *Successfull business models for 3D printing: Seizing opportunities with a game changing technology*. [Berenschot], [Utrecht], 2014.

32 Nicolas Serres, Dorian Tidu, Simon Sankare, and Françoise Hlawka. Environmental comparison of meso-clad® process and conventional machining implementing life cycle assessment. *Journal of cleaner production*, 19(9-10):1117–1124, 2011. doi:10.1016/j.jclepro.2010.12.010.

33 Ultimaker. Technical data sheet pla, 2018. URL: https://ultimaker.com/download/74970/UM180821%20TDS%20PLA%20RB%20V11.pdf.

34 Ultimaker Cura. 3d printing software for a powerful production workflow, 2019. URL: https://ultimaker.com/software.

35 Thao van Le, Henri Paris, and Guillaume Mandil. Environmental impact assessment of an innovative strategy based on an additive and subtractive manufacturing combination. *Journal of cleaner production*, 164:508–523, 2017. doi:10.1016/j.jclepro.2017.06.204.

36 VDI e.V. Vdi 3405: Additive fertigungsverfahren grundlagen, begriffe, verfahrensbeschreibungen, 2014.

37 E. Westkämper, Alting, and Arndt. Life cycle management and assessment: Approaches and visions towards sustainable manufacturing (keynote paper). *CIRP Annals*, 49(2):501–526, 2000. doi:10.1016/S0007-8506(07)63453-2.

38 Terry Wohlers and Robert Ian Campbell. *Wohlers report 2019-3D printing and additive manufacturing state of the industry*. Wohlers Associates, Fort Collins, Colo., 2019.

39 Li Yi, Moritz Glatt, Tai-Yen Thomas Kuo, Ankyd Ji, Bahram Ravani, and Jan C. Aurich. A method for energy modeling and simulation implementation of machine tools of selective laser melting. *Journal of cleaner production*, 263:121282, 2020. doi:10.1016/j.jclepro.2020.121282.

40 Li Yi, Bahram Ravani, and Jan C. Aurich. Development and validation of an energy simulation for a desktop additive manufacturing system. *Additive manufacturing*, 2019. doi:10.1016/j.addma.2019.101021.

41 YOKOGAWA. Wt 1800e high-performance power analyzer, 2019. URL: https://tmi.yokogawa.com/solutions/products/power-analyzers/wt1800e-high-performance-power-analyzer/.

A Appendix: NC code and database-driven simulation approach

By considering the AM machine as an entire system, its total energy consumption (E_{AM}) is the sum of the energy consumptions of the system components ($E_{Component}$) within the system; therefore, E_{AM} can be defined by the following equation:

$$E_{\mathrm{AM}} = \sum_{i=1}^{n} E_{\mathrm{Component\ i}} \tag{2}$$

For every system component, its energy consumption is the time integration of power ($P_{Component}$). Based on the trapezoidal rule, $E_{Component}$ in a given time (T_N) can be expressed as follows:

$$E_{\mathrm{Component}} = \int_0^{T_{\mathrm{N}}} P_{\mathrm{Component}} dt \approx \sum_{k=1}^{N} \frac{P_{\mathrm{k\text{-}1}} + P_{\mathrm{k}}}{2} (t_{\mathrm{k}} - t_{\mathrm{k\text{-}1}}) \tag{3}$$

Thus, based on the equations described above, the most important variables for enabling the energy simulation of AM are power data and time data. In the NC code and database-driven approach, the time data of a build task are collected from its NC code, and the power data are acquired from a power database, where the power data of system components are already stored. The functional logic of the NC code and database-driven approach is illustrated in Figure 9 and described below.

In the first step, the time parameters are read from the NC code, based on which a new time array is created in the second step. According to the information of the NC code, the times for different operations can be identified. For example, in Figure 9, it is assumed that the time array is made with a sampling time of 1 s and that during the 10 s for layer 1, 5 s

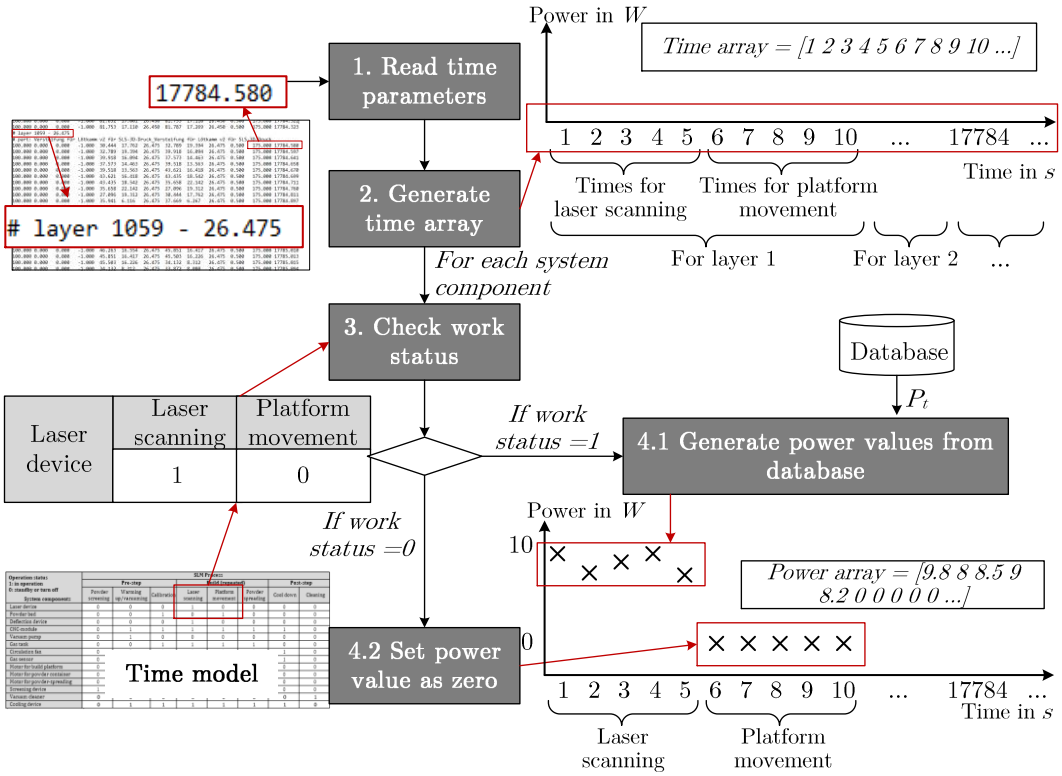


Figure 9 NC code and database-driven approach.

are used for the laser scanning, while the remaining 5 s are used for the platform movement. In the third step, it is to check the operation status of each system component. For example, Figure 9 shows the operation status of the laser device that it is in operation (denoted as "1") in the laser scanning and standby in the platform movement (denoted as "0"). In the fourth step, when the work status of a system system is "1" during an operation, the power data of this system component should be allocated to the times for this operation, while, when the work status is "0", the power data of this system component during the corresponding operation are set to zeros. For example, in Figure 9, it is known that the laser device is in operation for the laser scanning and standby for the platform movement; hence, for the laser scanning, the power values of the laser device are generated from the database (note that the values of 9.8, 8, 8.5 etc. are pseudo and do not represent the real power data), while during the platform movement the power values are zeros. By summarizing the power values, a power array for a system component with a same length as the time array can be created. Finally, by adopting Equation 3, the energy consumption of this system component can be calculated, in which the power and time array generated respectively by the database and NC code serve as $P_{Component}$ and T_N. Moreover, by adopting Equation 2, the total energy consumption of an AM system can be determined.

Physical Modeling of Process-Machine-Interactions in Micro Machining

Andreas Lange[1]
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Benjamin Kirsch
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Marius Heintz
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Jan C. Aurich
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Abstract

Increasing demands for smaller and smarter devices in a variety of applications requires the investigation of process-machine-interactions in micro manufacturing to ensure process results that guarantee part functionality. One approach is the use of simulation-based physical models. In this contribution, methods for the physical modeling of high-precision air bearing and magnetic bearing spindles are presented in addition to a kinematic model of the micro milling process. Both models are superimposed in order to carry out investigations of the slot bottom surface roughness in micro end milling. The results show that process-machine-interactions in micro manufacturing can be modeled by the superposition of a physical model of the machine tool spindle taking cutting forces into consideration and a purely kinematic model of the machining process, providing the necessary tools for a variety of further investigations into process-machine-interactions in micro manufacturing.

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases multiphysics, air bearing, magnetic bearing, surface roughness modeling, micro milling

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.2

Funding *Andreas Lange*: This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057.

1 Introduction

The increasing demand for smaller and smarter products in a wide range of applications poses new challenges for manufacturing technology [31]. Even the slightest influence on the machining process, whether machine or process induced, can harm the functionality of the component [7]. For this reason, the investigation of process-machine-interactions is moving into focus within the field of micro cutting [1]. The term process-machine-interaction (PMI) refers to the interaction between a manufacturing process and the machine tool used. These interactions can negatively influence the machining process and thus need to be investigated in order to minimize potentially harmful process-machine-interactions and also identify possible positive effects of certain process-machine-interactions on process results [8]. In

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 2; pp. 2:1–2:20

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

macro machining, numerous methods to analyze the effects of process-machine-interactions have been developed. A comprehensive overview of these methods can be found in [7] and [8]. Process-machine-interactions can affect both the machine tool, especially the machine tool spindle and the tool, and the process result. This can result in increased tool wear or premature tool breakage [8]. Also, process-machine-interactions can influence the process result and thus eventually affect the functionality of the part [13]. In micro machining, the dimensions of the tool and the chip to be removed are significantly smaller than in macro machining, which causes changes in the physical behavior during machining known as size effects [27], which is why existing methods for investigating process-machine-interactions in macro manufacturing cannot be adopted to micro manufacturing [1]. As such, the machine tool and machine tool spindle, including the tool, must be regarded as separate aspects [21]. While macro machining aims to reduce chatter vibrations by increasing the stiffness of the machine tool and using shorter or stiffer tools, an excessive influence of spindle run-out on the process result is observed in micro machining [21]. Thus, in order to use simulation-based physical models to investigate process-machine-interactions in micro manufacturing, detailed physical models of both the machine tool spindle including the tool, and the process, are required to capture process-machine-interactions.

Simulation-based investigations of micro cutting are either based on purely kinematic simulations [3, 23], chip formation simulations [2], or on kinematic simulations with added cutting force models [1, 22, 26]. Purely kinematic simulations model the ideal process kinematics and thus allow a determination of the shape of the chip to be removed, the slot geometry, and the ideal surface roughness in the slot bottom and slot flank [21]. However, these purely kinematic simulations do not consider the influence of machine vibrations and cutting forces and thus do not allow an investigation of process-machine-interactions [3, 23]. Chip formation simulations can be used to analyze the cutting energies, cutting forces and chip geometry under consideration of the cutting tool geometry and elasticity and plasticity effects of the workpiece material [2]. The difficulty lies in the need for a precise knowledge of the workpiece‘s elastic and plastic behavior [20]. Additionally, chip formation simulations require a high amount of computing time and resources [2]. Because of this requirement, usually only a fraction of the tool movement is modeled [25, 20], preventing the investigation process-machine-interactions. Kinematic simulations with added cutting force models are able to combine the advantages of purely kinematic simulations without omitting the influence of the cutting forces. As such, coupling these models and the kinematics of the machine tool is a promising approach to model process-machine-interactions in micro manufacturing and has already been implemented by several researchers [1, 26, 15]. As of now, the implemented machine tool kinematics are based on beam models of the tool, allowing the consideration of tool deflection but do not include the vibrational behavior of the spindle.

Due to the excessive influence of the machine tool spindle's behavior on the process result in micro machining [21], high-precision spindles for micro manufacturing are required. Because of the high requirements in terms of low friction, precision, stiffness, damping, and rotational speeds, air bearings and magnetic bearings are used almost exclusively. Various physical models for the investigation of air bearings have been presented [29, 19]. Most of these models are based on the Reynolds equation for thin-film fluid flows [24]. This Reynolds equation assumes an inertialess, laminar fluid flow. However, the accuracy of air bearing simulations modeled with the Reynolds equation suffers from neglecting inertia effects, sophisticated turbulence models and the analytical approximation of the orifice inlet mass flow rate [11]. Magnetic bearings require a multiphysical simulation model to allow data exchange between the coupled structural-electromagnetic-closed-loop-behavior [17]. Usually,

the electromagnetic behavior is modeled using the magnetic circuit method, which represents the magnetic bearing through a network of resistors [17]. The magnetic circuit method is based on the electromagnetic force calculation of a simple U-shaped magnet and thus not able to fully capture the electromagnetic behavior of complex geometries and nonlinear material behavior [17]. Detailed finite element models can capture the spindle's rotordynamic behavior considering gyroscopic effects and mass unbalance [19, 12]. But these rotordynamic models of high-precision spindles usually do not include the influence of tool clamping, tool geometry, and cutting forces, meaning that process-machine-interactions cannot be analyzed with these models [12].

The benefit of investigating process-machine-interactions using physical simulation models is the ability to identify the influences and effects of single parameters on components and the overall process result [8]. Furthermore, simulations can be used to quantify parameters relevant for the understanding of process-machine-interactions, which are difficult or impossible to determine experimentally [25]. Thus, physical models can be a valuable resource to deepen the understanding of tool behavior, machine vibrations and process behavior considering the process-machine-interaction. However, for a proper simulation-based investigation into process-machine-interactions in micro cutting, the aforementioned shortcomings must be overcome to achieve the modeling accuracy required to capture process-machine-interaction influences and effects. The objective of this paper is to present a suitable method for modeling process-machine-interactions in micro manufacturing. For this purpose, methods for the accurate physical modeling of machine tool spindles supported by air bearings and magnetic bearings, as well as methods for the kinematic modeling of the process kinematics are given. Further, a method for the superposition of machine and process kinematics is given to enable the investigation of process-machine-interactions under consideration of cutting forces.

2 Methods

As explained in section 1, modeling process-machine-interactions in micro manufacturing requires high-quality simulation models, which are able to capture the small-scale phenomena of micro machining. This section describes the methods used to model the machine kinematics with emphasis on the machine tool spindle dynamics as well as the process kinematics of micro end milling.

2.1 Machine kinematics

The overall kinematic behavior of machine tool spindles is mainly influenced by the spindle's bearings, the rotor's dynamics due to structural loads, the rotor's dynamics due to thermal loads, the tool clamping, and the tool (see also [21]). This necessitates a physical model of the machine tool spindle which is able to capture all of the aforementioned aspects.

2.1.1 Air bearing spindles

The physical model of an air bearing spindle can be set up by first setting up, simulating, and evaluating the fluid dynamics model of the air bearings and then using the results of this fluid dynamics model for the structural dynamics model of the rotor, tool clamping, and tool. For this purpose, the relevant bearing characteristics to be determined are the dynamic stiffness and damping coefficients. These bearing characteristics can be evaluated using the small perturbation method [6]. Herein, five cases of the fluid dynamics model must be set up. The first case is used to determine the equilibrium pressure distribution

(and thus fluid forces acting on the rotor) under the bearing's given static load, such as gravitational force. Subsequently, the rotor, which is in its equilibrium position, is perturbed to obtain induced restoring forces and hence dynamic characteristics of the bearing. Since the induced forces are directly proportional to the displacements and associated velocities, only four additional cases are necessary to determine the perturbed pressure distributions: a perturbation along the x-direction (case 2), a perturbation along the y-direction (case 3), a perturbation along the x-direction with an applied perturbation velocity in x-direction (case 4), and a perturbation along the y-direction with an applied perturbation velocity in y-direction (case 5). Once all five pressure distributions are known, the dynamic stiffness and damping coefficients can be calculated.

ANSYS Fluent 2019R3 and ANSYS Mechanical APDL 2019R3 were used to set up the models and conduct the simulations [2].

Bearing characteristics

Since the majority of air bearing spindles are equipped with inlet restricted air bearings, where pressurized air is fed into the bearing gap through a discrete number of orifices, this paper will only focus on this air bearing configuration. Further, isotherm fluid flow may be assumed [29] and air is modeled by the ideal gas law.

A schematic view of an aerostatic journal bearing with a feed orifice is given in figure 1. The geometric properties are given by the rotor radius R, the orifice radius r_0, the bearing length L, and the bearing gap c. Pressurized air with a predefined pressure p is forced into the bearing gap through a restricting orifice. The air flows through the bearing gap and exists the bearing at the rear and front face of the bearing ($x = L$), where ambient pressure p_a prevails. Due to unavoidable external excitations and manufacturing errors, the centers of the rotor and the bearing sleeve are never coincident, but are offset by small deviations in x- and y-direction (Δx, Δy). Hence, the bearing gap height h is a sum of the coaxial bearing clearance c (bearing gap height for a rotor and bearing sleeve perfectly coincident with respect to each other) and the eccentricity e. Consequently, the bearing gap height varies with respect to the circumference of the rotor.

The fluid flow within the bearing can be divided into three regions:

- the orifice region (I),
- the entrance region (IIa)) and
- the gap region (IIb)).

Within the orifice and entrance region, the abrupt redirection of the air flow leads to complex flow phenomena such as turbulence and eddies [9]. In the gap region, these complex flow phenomena have already subsided, which leads to a viscous/laminar flow [29]. The corresponding qualitative pressure profile within an aerostatic journal bearing is depicted by the blue line in figure 1. A detailed representation of the pressure distribution, including the pressure drop in the entrance region (IIa) is required for the accurate computation of the rotors' equilibrium position, the equilibrium pressure distribution and hence for the computation of the static and dynamic characteristics.

For the solution of the equilibrium pressure distribution and with respect to the different behavior within the bearing regions I and II, the fluid dynamics model of an aerostatic journal bearing can be split up into two submodels, which are then coupled through an

[2] Naming of specific manufacturers is done solely for the sake of completeness and does not necessarily imply an endorsement of the named companies nor that the products are necessarily the best for the purpose.

iterative solution approach at the interface depicted in red in figure 1. The split up submodels are shown in figure 1. Even though both submodels must be solved during every iteration, dividing the aerostatic bearing into two submodels is computationally more efficient compared to the use of a single computational domain. This is due to the two submodels not requiring the meshing and computation of multiple three-dimensional orifice regions.

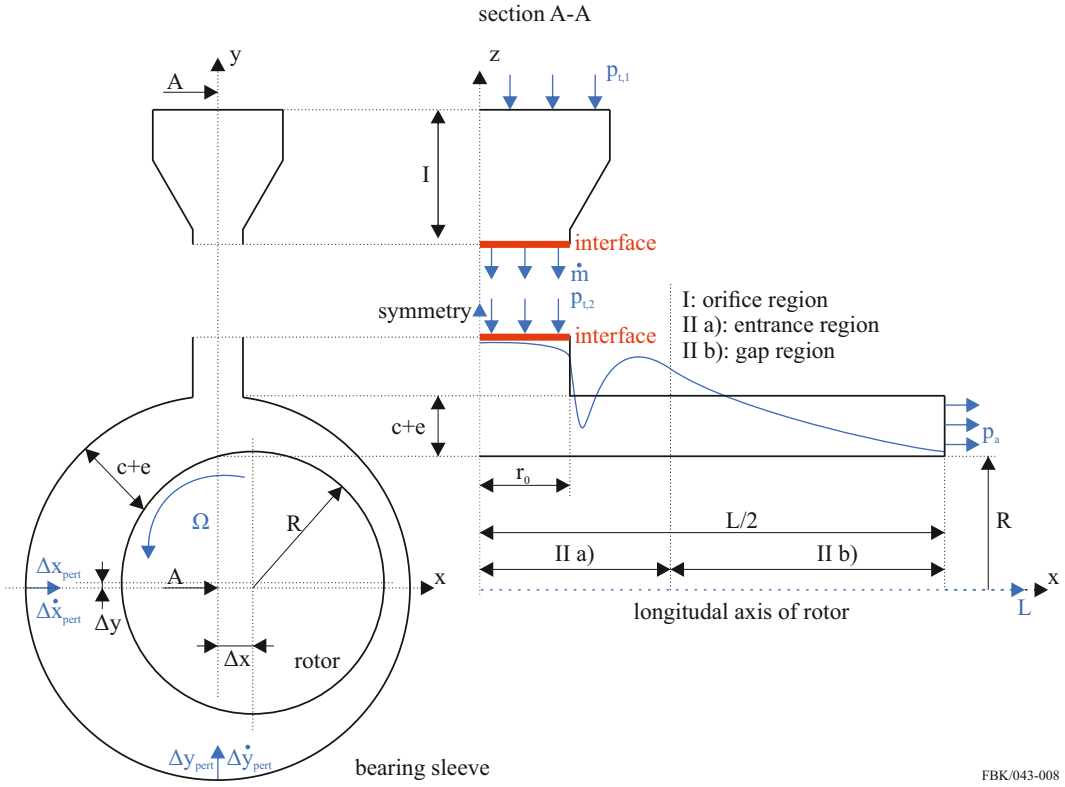


Figure 1 Schematic view of an aerostatic journal bearing with qualitative pressure profile in the individual regions.

Submodel I consists of the orifice region (I). Taking advantage of the symmetric geometry of the orifice, a two-dimensional simulation model can be set up by applying a rotational symmetry boundary condition to the orifice centerline. The boundary conditions of this submodel are specified by applying a predefined supply pressure $p_{t,1}$ at the orifice inlet and a mass flow rate $\dot{m}$ at the interface (the orifice region's outlet). The exact value of the mass flow rate is yet unknown. The governing equations for this submodel are the Reynolds-averaged Navier-Stokes equations for axisymmetric geometries [5]. Turbulence is modeled using the Menter-k-ω-SST model [30].

Submodel II consists of the entrance region (IIa)) and the gap region (IIb)). Because of the varying gap height, a three-dimensional simulation model is required and only one symmetry boundary condition can be applied at the y-z-plane. The boundary conditions of this submodel are specified by applying an interface pressure $p_{t,2}$ at the interface (entrance region's inlet). This interface pressure is dependent on the prior solution of submodel I and as such yet unknown. At the bearing outlet, an ambient pressure p_a is applied. The rotational speed boundary condition is applied to the rotor. The fluid flow in submodel II can be modeled using the Reynolds-averaged Navier-Stokes equations [5].

Using an iterative solution approach, both submodels can be coupled to determine the correct mass flow rate and interface pressure. In order to determine suitable initial conditions for the mass flow rate m_{init} and interface pressure, an analytical formula for the calculation of the mass flow rate through the orifice may be used. This formula is given by [28]:

$$\dot{m}_{init} = C_D \cdot 2\pi r_0 c \cdot \Psi \left(2 \cdot p_i \cdot \frac{p_i}{R_S T_{abs}} \right) \tag{1}$$

where C_D is the discharge coefficient [28], R_S is the specific gas constant, $T_{abs}$is the absolute temperature, and Ψ is the discharge function [28] of the fluid.

With the analytical formula, both submodels, and the defined interface boundary conditions, the following algorithm can be employed to determine the stationary pressure distribution in the bearing:

1. Analytical calculation of initial mass flow rate $\dot{m}_{init}$ value according to equation 1
2. Simulation of orifice region using determined mass flow rate
3. Determination of total pressure $p_{t,2}$ at orifice outlet
4. Simulation of entrance and gap regions using $p_{t,2}$ as the inlet boundary condition
5. Determination of mass flow rate $\dot{m}$ evaluated at entrance and gap regions
6. Repeat steps 2 to 5 until convergence criteria are met

For aerostatic bearings, suitable convergence criteria can be established by monitoring the scaled residuals of the Reynolds-averaged Navier-Stokes equations and the fluid forces acting on the rotor in x- and y-direction. Convergence can be assumed as soon as the fluid forces remain constant and the scaled residuals either drop by four orders of magnitude or remain constant. Convergence should not be judged by just monitoring the scaled residuals of Reynolds-averaged Navier-Stokes without monitoring the fluid forces.

Once the convergence criteria are met, the first of the aforementioned five cases is solved and the equilibrium pressure distribution is obtained. For the remaining four cases, the mass flow rate and pressure $p_{t,2}$ can be reused. Hence, only region II must be recomputed.

The remaining four cases are set up similar to the first case, the only difference being an additional boundary condition applied to the bearing sleeve to account for the perturbation defined by the perturbation increment. This boundary condition is applied as follows:

- case 2: an additional translational offset Δx_{pert} is applied to the bearing sleeve,
- case 3: an additional translational offset Δy_{pert} is applied to the bearing sleeve,
- case 4: an additional translational offset Δx_{pert} and a translational velocity $\Delta \dot{x}$ is applied to the bearing sleeve,
- case 5: an additional translational offset Δy_{pert} and a translational velocity $\Delta \dot{y}$ is applied to the bearing sleeve.

Once all five cases are solved, the pressure distributions can be integrated to obtain the fluid forces acting on the rotor. Using the small perturbation method, a first order Taylor series expansion yields:

$$F_{x,i} = F_{x,1} + k_{xx}\Delta x + k_{xy}\Delta y + c_{xx}\Delta \dot{x} + c_{xy}\Delta \dot{y}, \qquad i = 2,3,4,5 \tag{2}$$

$$F_{y,i} = F_{y,1} + k_{yx}\Delta x + k_{yy}\Delta y + c_{yx}\Delta \dot{x} + c_{yy}\Delta \dot{y}, \qquad i = 2,3,4,5 \tag{3}$$

It can be seen from equations 2 and that no additional case with superpositioned perturbations along both the x- and y-direction is necessary. For example, in case 2, a single perturbation along the x-direction gives a perturbation of zero in y-direction and zero perturbation velocity, which cancels out the cross-coupled stiffness term $k_{xy} \cdot \Delta y$ and the damping terms and hence enables the computation of the direct stiffness k_{xx}. The same principle holds for case 3. For case 4, a perturbation along the x-direction and the application of a perturbation

velocity $\Delta\dot{x}$ again cancels out the cross-coupled stiffness term $k_{xy} \cdot \Delta y$ and the damping term $c_{xy} \cdot \Delta y_{pert}$. Using the result of case 2, the direct stiffness k_{xx} is already known, and thus the direct damping c_{xx} can be computed. where k and c are called linearized stiffness and damping coefficients respectively. Once again, the same principle holds for case 5. Finally, the individual stiffness and damping coefficients can be computed as:

$$\begin{pmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{pmatrix} = \begin{pmatrix} \frac{F_{2,x}-F_{1,x}}{\Delta x} & \frac{F_{3,x}-F_{1,x}}{\Delta y} \\ \frac{F_{2,y}-F_{1,y}}{\Delta x} & \frac{F_{3,y}-F_{1,y}}{\Delta y} \end{pmatrix}, \qquad \begin{pmatrix} c_{xx} & c_{xy} \\ c_{yx} & c_{yy} \end{pmatrix} = \begin{pmatrix} \frac{F_{4,x}-F_{1,x}}{\Delta \dot{x}} & \frac{F_{5,x}-F_{1,x}}{\Delta \dot{y}} \\ \frac{F_{4,y}-F_{1,y}}{\Delta \dot{x}} & \frac{F_{5,y}-F_{1,y}}{\Delta \dot{y}} \end{pmatrix} \quad (4)$$

where the first index of Force F depicts the case number which the force is evaluated in and the second index depicts whether the force acts in x- or y-direction.

Rotordynamics

The general dynamic equation to describe a system in motion is given by [10]:

$$[\mathbf{M}]\left\{\ddot{U}\right\} + [\mathbf{C}]\left\{\dot{U}\right\} + [\mathbf{K}]\{U\} = f \quad (5)$$

where $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are the mass, damping and stiffness matrices, and f is the external force vector. In rotordynamics, this equation gets additional contributions from the gyroscopic effect [G], and the rotating damping effect [B] which leads to the modified equation [10]:

$$[\mathbf{M}]\left\{\ddot{U}\right\} + ([\mathbf{G}] + [\mathbf{C}])\left\{\dot{U}\right\} + ([\mathbf{B}] + [\mathbf{K}])\{U\} = f \quad (6)$$

Equation 6 is formulated using a stationary reference frame. The bearing coefficients, calculated with the methods in section 2.1.1 are incorporated in the matrices $\mathbf{K}$ and $\mathbf{C}$. The external force vector f contains the mass unbalance forces, cutting forces and may also include additional external forces, such as magnetic bearing forces (section 2.1.2).

2.1.2 Magnetic bearing spindles

The physical model of magnetic bearing spindles consists of the calculation of the electromagnetic forces and the setup of the rotordynamics model. For the required data exchange during the whole simulation, both aspects must be included in the closed loop model used for the control system analysis. In this section, a method is introduced, where the electromagnetic forces are first obtained using a three-dimensional electromagnetic finite element method (FEM) simulation model. These forces are then stored within lookup-tables to represent the behavior of a simulation model within the control loop.

Evaluation of electromagnetic forces

A schematic view and the applied boundary conditions of a radial heteropolar 4-pole-pair active magnetic bearing are shown in figure 2.

A pole configuration of alternating north (N) and south (S) poles (depicted by the N and S symbols in two of the eight poles in figure 2a)) and a differential winding scheme is employed. Herein, one electromagnet is operated with the sum of the bias current i_0 and the control current i while the opposite electromagnet uses the difference (figure 2b)). This ensures that the current and thus the force increase in one electromagnet to the same extent as they decrease in the other electromagnet. Depending on the material used, both the rotor and the iron core can exhibit magnetic saturation, where an increase of the applied external magnetic field H does not significantly increase the magnetic flux density B within

the material [17]. Regarding the magnetic bearing shown in figure 2, this means that once the material is saturated, an increase of the currents through the coils does only lead to a neglectable change in the electromagnetic forces. This nonlinear behavior is included in the model through so-called B-H-curves [17]. The required boundary conditions for the three-dimensional electromagnetic simulation are the number of turns, the conducting area, the current flow direction and the current value for every coil. An edge-based magnetic vector potential method based on the Maxwell equations [14], is used for the solution and evaluation of the magnetic flux distribution and magnetic forces. The electromagnetic simulation model computes electromagnetic forces based on predefined current inputs. However, the magnetic forces also depend on the rotor position. Hence, the model must be parametric regarding the rotor's position in x- and y-direction to fully capture the dependencies $f_{mag,x} = f(x, y, i_x, i_y)$ and $f_{mag,y} = f(x, y, i_x, i_y)$ to generate accurate lookup-tables.

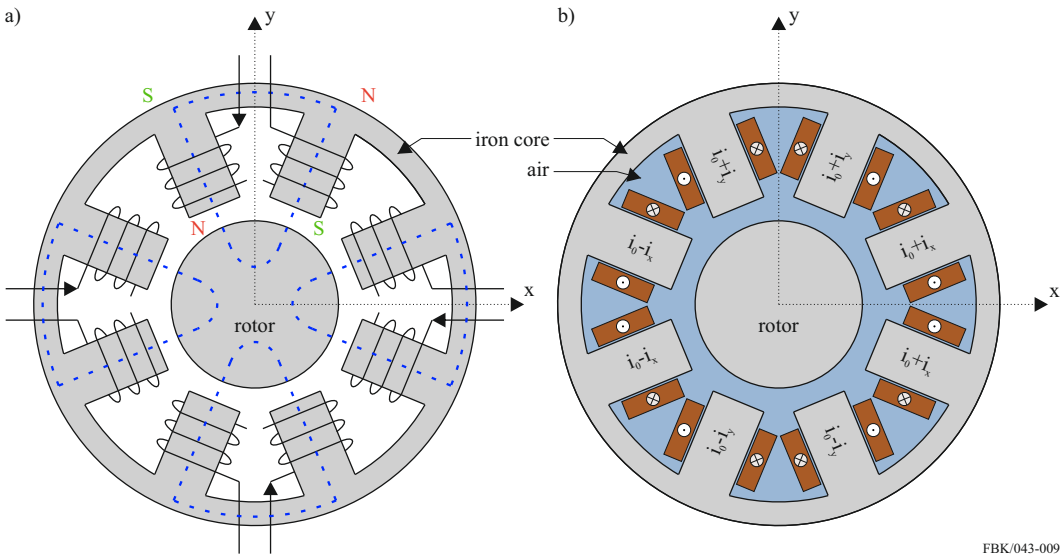


Figure 2 a) schematic view and b) boundary conditions of a radial heteropolar 4-pole-pair active magnetic bearing.

Rotordynamics

The magnetic bearing spindle's rotordynamics can be described using either equation 5 or equation 6. Compared to the rotordynamics of the air bearing spindle, where the bearing characteristics are inserted into the left-hand-side of the dynamics equation, in the rotordynamics model of the magnetic bearing spindle the bearing forces are applied as additional external forces within the force vector f, acting on the rotor.

Control system

Simulink R2019b [3] is used for the setup of the control system model. Distributed proportional-integral-derivative (PID) control is used for the current control of the magnetic bearing.

[3] Naming of specific manufacturers is done solely for the sake of completeness and does not necessarily imply an endorsement of the named companies nor that the products are necessarily the best for the purpose.

Herein, each controlled variable (currents i_x, i_y) is assigned an input (positions x, y). A single-variable PID controller is designed and implemented for each pair of inputs and outputs. Implementing independent PID controllers for the current control in both x- and y-direction, the dynamics equations (equation 5 or 6), and the magnetic bearing characteristics through the described lookup-tables, one obtains a model of the closed loop, which can be used for the system simulation and the controller design. The closed loop can be seen in figure 3a), with a detailed schematic of the plant given in figure 3b).

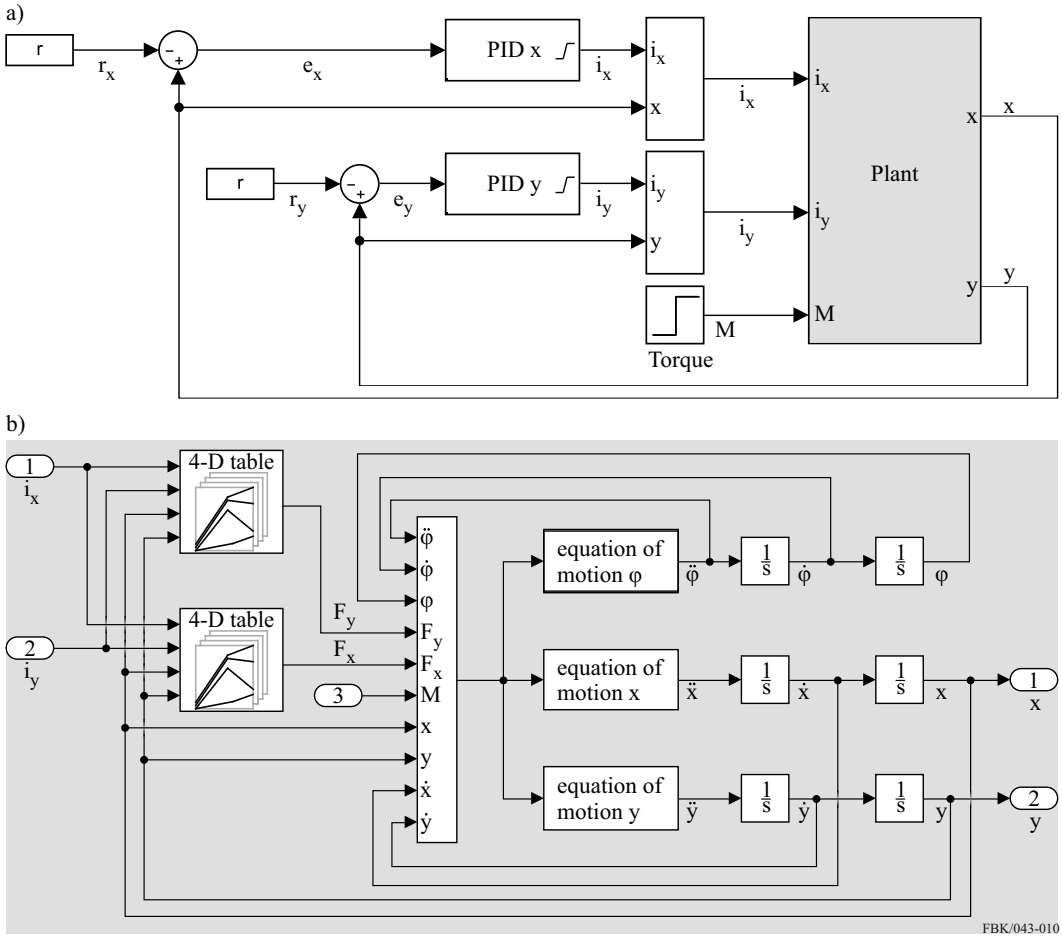


Figure 3 Control system for a spindle rotor supported by two radial heteropolar 4-pole-pair active magnetic bearings.

The model dynamics can be described as follows: During a time step Δt and for a given target position r_x and r_y of the spindle rotor, the PID controllers determine the deviations e_x and e_y from the instantaneous positions x and y and accordingly alter the control currents i_x and i_y. The updated control currents get passed to the plant subsystem (figure 3b)) along with the instantaneous positions x and y of the spindle and the torque of the spindle's driving motor. Within the plant subsystem, the electromagnetic force $f_x, f_y = f(x, y, i_x, i_y)$ is calculated by looking up its x- and y-value corresponding to the instantaneous position and currents within the four-dimensional lookup table. The updated electromagnetic force is then transferred to the rotordynamics model. In order to solve the rotordynamics model, additional data is required. This additional data consists of the instantaneous values of

angular acceleration $\ddot{\varphi}$, angular velocity $\dot{\varphi}$, rotational angle φ, positions x and y and velocities $\dot{x}$ and $\dot{y}$. When all data is evaluated, the rotordynamics model is solved numerically to obtain the accelerations $\ddot{x}$ and $\ddot{y}$. Accelerations are then integrated twice to obtain both updated instantaneous velocities and positions. With the updated positions x and y, the next time step can be computed.

2.2 Process kinematics

The setup of the simulation model for the determination of the process kinematics will be explained with respect to a single-cutting-edge micro end mill with an infinitely sharp cutting edge, as shown in figure 4.

The ideal process kinematics of micro end milling can be modeled by superimposing the translational tool feed movement with a rotational movement of the tool. Without the influence of machine kinematics and a cutting edge radius of zero, the kinematics of a micro milling process with a single-cutting-edge micro end mill can be described by the following equation:

$$\begin{pmatrix} x_P \\ y_P \\ z_P \end{pmatrix} = \begin{pmatrix} r \cdot \sin(\varphi) \\ r \cdot \cos(\varphi) \\ (R-r) \cdot \tan(\chi_r') \end{pmatrix} + \begin{pmatrix} \frac{f_z}{2\pi} \cdot \varphi \\ 0 \\ 0 \end{pmatrix} \quad r \in [0, R] \tag{7}$$

Herein, the tool feed rate of the tool with a radius R is defined by the feed per tooth f_z along the positive x_P-direction. The slope of the cutting edge is reflected by the height function $f(z_P) = (R-r) \cdot \tan(\chi_r')$ along the z_P-direction.

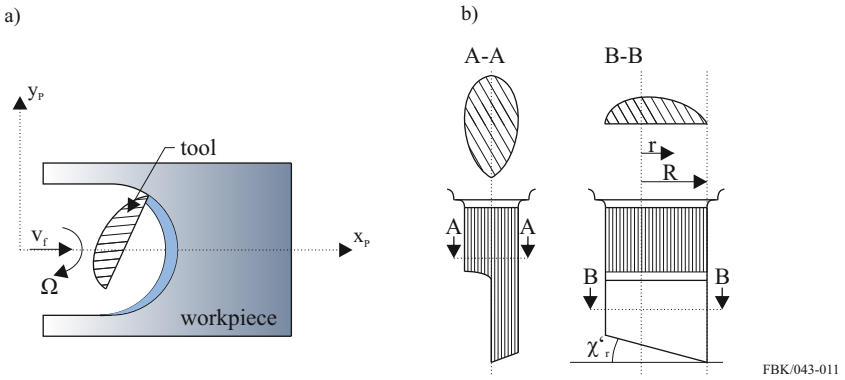


Figure 4 Schematic view of a micro end milling process (a) with a single-cutting-edge tool (b).

2.3 Process-machine-interaction

In the final physical model, the superposition of the machine kinematics and the ideal process kinematics yields the kinematics under consideration of process-machine-interactions. The superposition of the machine and process kinematics is achieved by using the following equation:

$$\begin{pmatrix} x_{PMI} \\ y_{PMI} \\ z_{PMI} \end{pmatrix} = \begin{pmatrix} x_P \\ y_P \\ z_P \end{pmatrix} + \begin{pmatrix} \cos(\varphi) & sin(\varphi) & 0 \\ \sin(\varphi) & -\cos(\varphi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_M \\ y_M \\ z_M \end{pmatrix} \tag{8}$$

This instantaneous position of the tool due to the process kinematics is described by three coordinates x_P, y_P, and z_P, where the P-index stands for process. The position of the tool center point with respect to the machine tool spindle was described by three coordinates x, y, and z in prior sections. For a better understanding, an M-index is added to these coordinates in the following sections, where M stands for machine.

Even though both the process and machine kinematics are modeled using a stationary reference frame, a rotation matrix is required for the correct superposition of both kinematics. The reason for the need of this additional multiplication lies within the nature of the rotordynamics model. Herein, the rotor is not actually spinning but only the physical effects, namely the gyroscopic effect and the rotating damping effect, are taken into account during the formulation of the rotordynamics equation (equation 6). Thus, the actual spin of the tool is accounted for by multiplying the machine kinematics x_m, y_m, and z_m with the rotation matrix about the rotational angle φ.

3 Results

Results will be presented in the following order: First, simulation results of the machine simulation model will be presented. Second, simulation results of the process simulation model and process-machine-interaction will be presented.

3.1 Air bearings

Results in this section will be reported for a single air bearing with the geometry properties and boundary conditions listed in table 1.

Table 1 Geometry properties and boundary conditions used for the analysis.

	parameter	value
geometry	bearing length	$70\,mm$
	rotor diameter	$21\,mm$
	number of orifices	6
	orifice radius	$0.09\,mm$
	bearing gap	$20\,\mu m$
	eccentricity	$3.5\,\mu m$
	perturbation increment	$1\,\mu m$
fluid properties	dynamic viscosity	$18\,\mu Pa \cdot s$
	orifice inlet temperature	$288.15\,K$
	orifice inlet pressure	$6\,bar$
	ambient pressure	$1\,bar$

Using the solution approach presented in section 2.1.1, one obtains the results for the mass flow rate as $\dot{m}_{iterative} = 4.211 \cdot 10^{-5}\,kg/s$. For comparison, the analytically calculated mass flow rate (equation 1) is given as $\dot{m}_{analytic} = 3.433 \cdot 10^{-5}\,kg/s$. Compared to the analytical value, the numerically calculated final value shows an increase of approximately 18.5%. Using the final mass flow rate and derived interface pressure, one can obtain the corresponding stiffness and damping coefficients as described in section 2.1.1.

3.2 Magnetic bearings

Results in this section will be reported for a rotor supported by two heteropolar 4-pole-pair active magnetic bearings. Geometry properties and boundary conditions are listed in table 2.

Table 2 Geometry properties and boundary conditions used for the analysis.

	parameter	value
geometry	rotor mass	$3.6\,kg$
	rotor moment of inertia about z-axis	$7.01 \cdot 10^{-5}\,kg \cdot m^2$
	rotor diameter	$43\,mm$
	iron core diameter	$100\,mm$
	number of poles	8
	number of coil turns per coil	460
	conducting area per coil	$42\,mm^2$
	bearing gap	$0.5\,mm$
bearing properties	bias current	$2.5\,A$
	control current	$\pm 2.5\,A$
	proportional gain	$10,000\,A/m$
	integral gain	$15,000\,A/ms$
	differential gain	$6\,As/m$
additional properties	simulation time	$4\,s$
	step size	$1 \cdot 10^{-4}\,s$
	solver	ode3
	target position in x_M- and x_M-direction	$0\,mm$

The control system model (figure 3) is used to simulate the time-dependent vibrations of the rotor. Figure 5 shows the corresponding vibration amplitudes, angular velocity and driving torque. During the simulation time of $4\,s$, the rotor's motion can be divided into three stages.

The first stage is labeled the lift-off phase in figure 5, ranging from $0\,s$ to $1\,s$. During this phase, the rotor is brought into levitation. During this stage, no driving torque is applied to the rotor and thus the rotor does not rotate. As can be seen in figure 5, the rotor is brought up from its initial resting position at approximately $3.5 \cdot 10^{-4}\,m$ to its target position of $x_{M,target} = y_{M,target} = 0\,m$. With the PID controller values of table 2, a maximum overshoot of approximately 23% is observed, followed by additional smaller oscillations until the rotor approaches its target position with an almost constant negative incline.

The second phase is labeled the acceleration phase. During this phase, a constant driving torque of $2\,Nm$ is applied to the rotor, generating a constant slope of the angular velocity up until $4000\,rad/s$ $(38,200\,min^{-1})$.

The third stage is labeled the stationary phase. Once the rotor has reached its nominal speed of $4,000\,rad/s$ no more driving torque is applied, leading to a constant rotor speed. In this phase, in addition to the mass unbalance, cutting forces are applied on the tool tip and applied once the simulation time reached $3\,s$.

The magnitude of the cutting force f_{cut} was set equal to $1\,N$, based on experimental investigations with comparable process parameters [16]. For a single-cutting-edge micro end mill, the cutting force can be assumed as a harmonic force with an excitation frequency ν equal to the angular velocity $\dot{\varphi}$. Since no influence of changing cutting conditions is considered (such as depth of cut being lower than the minimum chip thickness), the cutting force was superimposed with an additional random white noise f_{noise} to introduce random changes in the cutting force.

The shift of $1\,s$ between the rotor reaching its nominal speed and the application of cutting forces was implemented in order for remaining oscillations of the acceleration phase to subside. Figure 5 shows that the magnetic bearings are able to stabilize the rotor despite the excitation by mass unbalance and cutting forces.

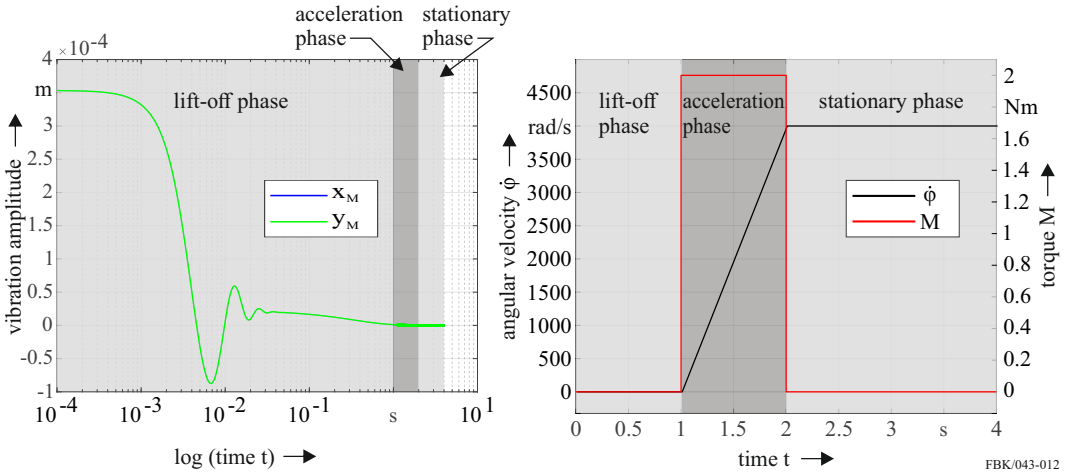


Figure 5 Vibration amplitudes, angular velocity and driving torque for a rotor supported by two radial heteropolar 4-pole-pair active magnetic bearings.

3.3 Process-machine-interaction

Ideal process kinematics results in this section will be shown for the process parameters listed in table 3. Process-machine-interaction results will be shown for the same process parameters under consideration of the machine kinematics determined by the physical model of the magnetic bearing spindle (see section 3.2).

Table 3 Process parameters used for the analysis.

parameter	value
effective tool radius	$25\,\mu m$
feed per tooth	$4\,\mu m$
cutting depth	$2.1\,\mu m$
minor cutting edge angle	$12°$
rotational speed	$38,200\,min^{-1}$

Using the process parameters listed in table 3, one can obtain the tool tip path trajectory in x_P-direction, y_P-direction, and x_P-y_P-direction for ideal process kinematics (no process-machine-interactions) and under consideration of process-machine-interactions as a function of time or absolute angular tool position as shown in figure 6. Figure 6a) and c) show the $x_P(t)$- and $y_P(t)$-trajectories of the tool tip (radius of $R = 25\,\mu m$) without the consideration of process-machine-interactions. As can be seen, the rotational movement of the tool tip leads to a sinusoidal tool path trajectory. Since the tool feed movement is modeled along the x_P-direction, the absolute amplitude of the sinusoidal tool path trajectory in x_P-direction is steadily rising due to the applied tool feed rate. In figure 6b), d), and f), the ideal process kinematics are superpositioned with the machine kinematics as described in section 2.3.

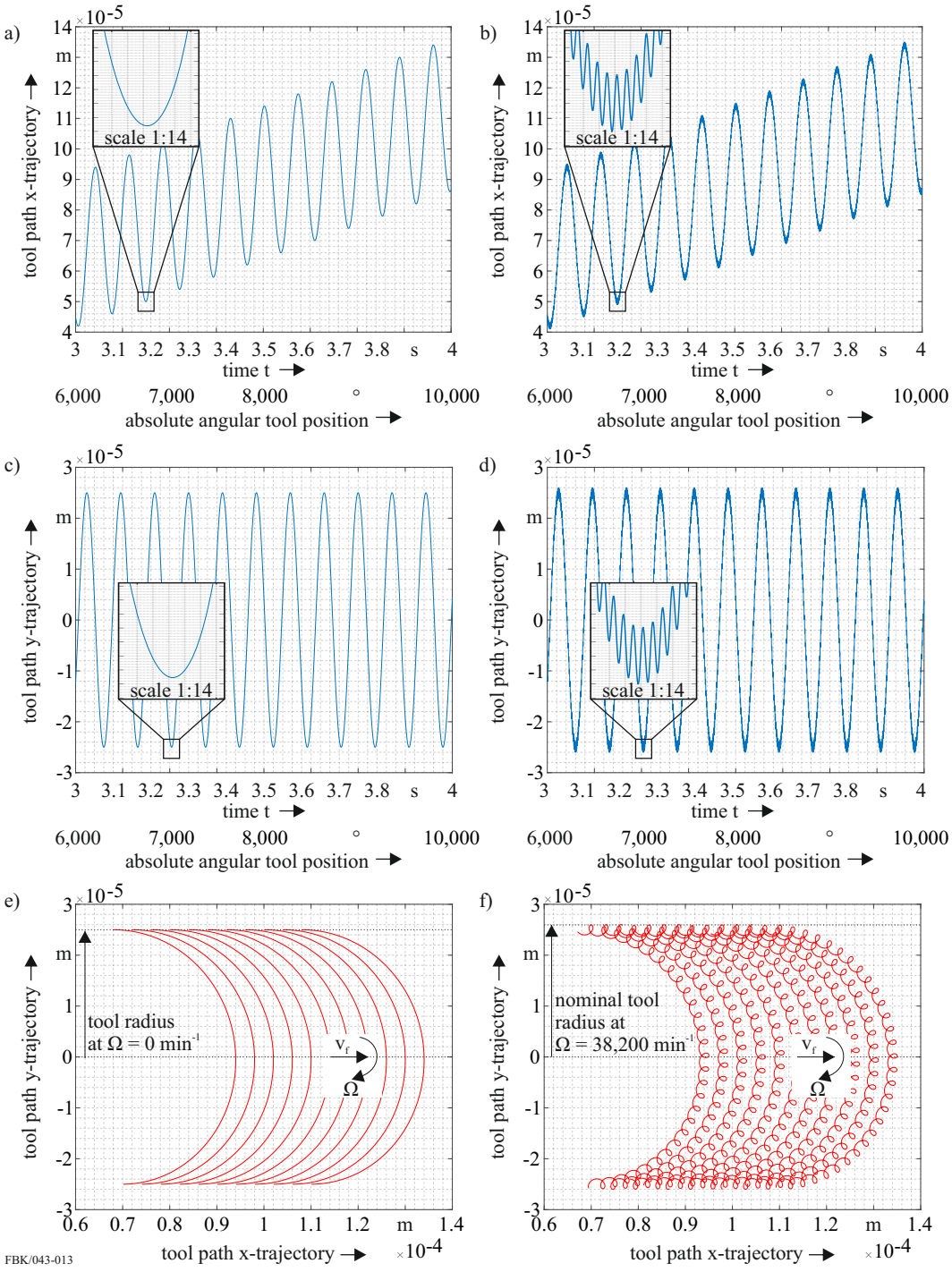


Figure 6 Tool tip path trajectory in x_P-direction, y_P-direction and x_P-y_P-direction for ideal process kinematics (a), c), e)) and under consideration of process-machine-interactions (b), d), f)).

Comparing the trajectories with and without process-machine-interactions against each other, it can be noted that the tool path trajectory under consideration of process-machine-interactions exhibits the same overall behavior as the ideal tool path trajectory but shows additional continuous high-frequency-oscillations with a small amplitude overlapping the ideal tool path trajectory.

Plotting the tool path trajectories in x_P- and y_P-direction (figure 6a) and c)) and figure 6b) and d) respectively) against each other gives the graphs depicted in figure 6e) and f). It must be noted that in these figures, only the front cutting motion is shown.

The aforementioned high-frequency-oscillations can be seen in figure 6f) in terms of loops which superimpose the ideal movement of the tool. In figure 6e), the ideal tool radius equals $R = 25\,\mu m$, whereas figure 6f) shows the nominal tool radius at a rotational speed of $38,200\,min^{-1}$. This nominal tool radius equals approximately $R_{nominal} = 27\,\mu m$, showing that the process conditions, in this case cutting width, do change under consideration of machine kinematics.

A three-dimensional surface representation and the surface profile of the ideal and process-machine-interaction-considering tool path trajectory is presented in figure 7. Unlike figure 6, the surface plots in 7a) and b) depict the trajectory of the whole tool, not just the tool tip. Further, visualizing the tool path trajectory along the z_P-axis picks up on equation 7, showing the effects of the height function $f(z_P)$ used to model the cutting edge.

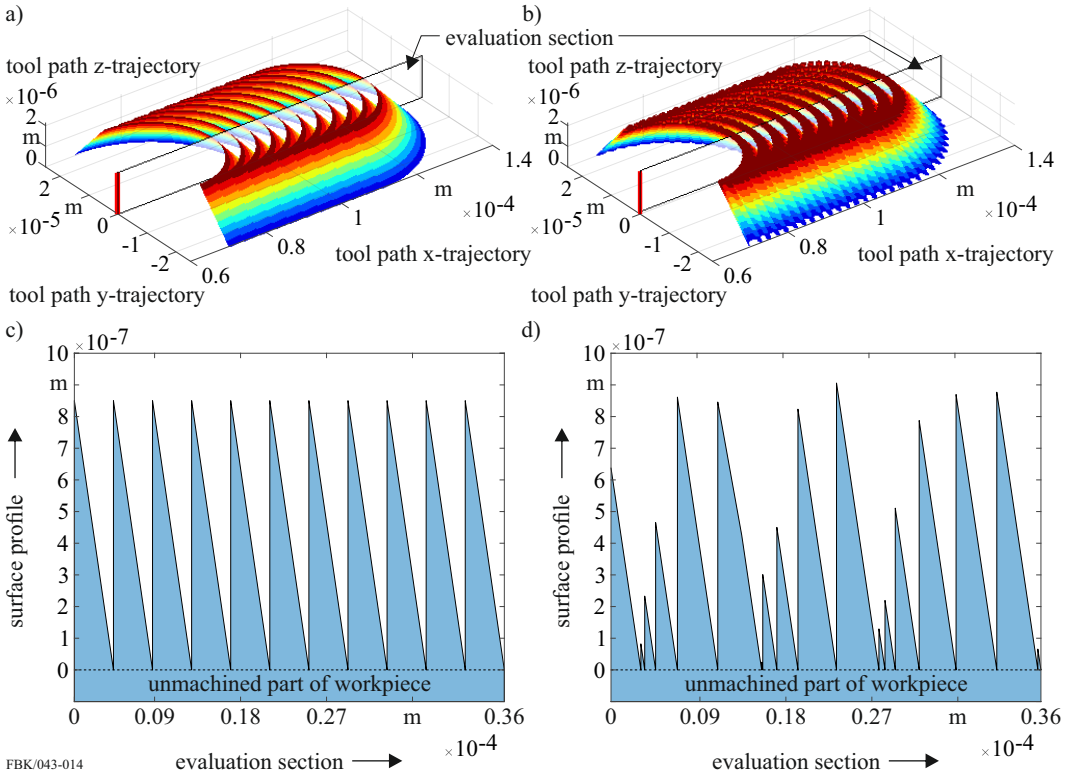


Figure 7 Three-dimensional view of tool path trajectory and filtered workpiece surface profile visualization for ideal process kinematics (a), c)) and under consideration of process-machine-interactions (b), d)).

Figure 7a) and b) also shows a graphical representation of the slicing plane used for the evaluation of the surface profile and roughness. The evaluation section of the surface profile visualizations in figure 7c) and d) is based on this slicing plane.

The unfiltered surface profile is filtered by evaluating the incline between all consecutive data points and comparing these inclines to the height function $f(z_P)$ of the minor cutting edge (see section 2.2 and equation 7). Excluded from this filtering are those data points that immediately follow the maximum values, since these data points form a vertical line that represents the vertical tool flank (see figure 4).

The filtered surface profile without and with consideration of process-machine-interactions is shown in figure 7c) and d). Regarding the surface profile without consideration of process-machine-interactions (figure 7c)), the graph shows a surface profile with equidistant and equal local minimum and maximum values as well as constant inclines between all local minima and maxima. The maximum surface roughness value equals $0.850\,\mu m$, and thus only about 40% of the cutting depth. The R_a-value is calculated by integrating the area between the surface roughness and the x_P-axis and equals $0.425\,\mu m$.

Regarding the surface profile with consideration of process-machine-interactions (figure 7d)), the surface profile exhibits big differences compared to the surface profile in 7c). As such, the local minima and maxima are neither equidistant nor equal anymore. The maximum surface roughness value equals $0.906\,\mu m$ and thus is 6% higher than the maximum surface roughness value without consideration of process-machine-interactions. Further, the R_a-value slightly increases to a value of $0.460\,\mu m$.

4 Discussion

This section discusses the presented methods and results in terms of application possibilities and compares them against other published work.

4.1 Air bearings

The presented model for the physical modeling of the pressure distribution of aerostatic journal bearings differs from existing models in multiple ways.

The implementation of the orifice region (I) into the physical model allows to develop a deeper understanding of the interactions between orifice region and entrance region, as was stated as an objective for future air bearing designs in [11].

The use of the coupled model omits the need to guess or approximate a discharge coefficient as was done in [29, 18]. Herein, the discharge coefficient was computed for an aerostatic thrust pad with a single orifice in the middle of the pad. The presented model uses the actual geometry of the radial bearing, including the orifice positions and surface curvature of both the rotor and the stator. Comparing the mass flow rate used in various approaches [29, 18] against the numerically calculated value using the presented method, it becomes clear that the consideration of the actual radial bearing geometry and configuration influences the mass flow rate (18.5% in the presented results), which will eventually influence the simulation of the process-machine-interactions.

The use of the Reynolds-averaged Navier-Stokes equations instead of the Reynolds equation offers the possibility to include inertia effects, turbulence and flow velocity and direction effects. Specifically, this enables the determination of the pressure distribution of aerostatic bearings featuring angled injection orifices similar to the physical model presented in [32]. However, with the model presented in this paper, the dynamic damping coefficients can be calculated, which eventually allows for a more accurate rotordynamics model of air bearing spindles, providing a way for the investigation into new and improved air bearing designs for air bearing spindles, such as bearings with angled injection orifices.

Drawbacks of the use of the Reynolds-averaged Navier-Stokes equations instead of the Reynolds equation are the increase in required computer resources, an increase in modeling time as well as an increased time consumption to compute the pressure distribution and the dynamic characteristics. Further, the presented method exhibits a lower robustness, which can lead to convergence difficulties.

4.2 Magnetic bearing spindles

Using a finite element model for the evaluation of the electromagnetic forces instead of the magnetic circuit method has the following advantages:

Nonlinear behavior such as magnetic saturation can be incorporated into the analysis, which is not possible using the magnetic circuit method.

No approximation of the electromagnetic forces using analytical formulas is required, leading to an overall more accurate close loop model and eventually to a more accurate model of the process-machine-interactions.

Implementing the numerically evaluated electromagnetic forces through lookup tables in the close loop model offers the possibility to include accurate properties of the magnetic bearings while keeping simulation time (and thus controller design and optimization time) as small as possible since no finite element analysis must be employed during each time step.

It is important to also mention the limits of the presented model.

Suitable lookup table data points have to be chosen in order to generate a reliable lookup table.

Every single lookup table data point has to be generated using the finite element model, which is more time consuming than using analytical formulas.

With respect to the time consumption of the finite element model to solve all necessary data points used for the presented results in section 3.2, the authors' assumption is that for greater number of inputs (> 4) as well as greater ranges of inputs (> 15 values to represent the behavior of every single input) the presented approach is not feasible anymore.

4.3 Process-machine-interaction

The presented approach for the physical modeling of process-machine-interactions offers a feasible approach to analyze process-machine-interactions in micro manufacturing. Combining a purely kinematic mathematical model of the ideal process kinematics with a rotordynamics model of the machine tool spindle under consideration of the cutting forces offers the possibility to capture the influence of the cutting forces on the machine‘s motion as well as the influence of the machine´s motion on the tool path trajectory.

Compared against several existing approaches, the presented approach offers several advantages.

The rotordynamics model is set up in a modular way. This allows for a straight-forward implementation of additional relevant aspects that influence the machine tool spindle‘s behavior (see section 2.1) and were not yet included in the analyses presented in section 3.

Even though the ideal process kinematics are modeled using a purely kinematic approach similar to approaches in [3, 23], cutting forces can be included within the rotordynamics model of the machine simulation model, providing a suitable approach to model process-machine-interactions.

The modular setup of both the machine and process models offers great flexibility. As such, instead of the micro end milling process analyzed in this paper (section 2.2), the process simulation model can be adapted to represent other processes, such as micro grinding [23]. Similarly, the machine simulation model can be adapted by changing the rotor and/or tool geometry to represent other machines and/or tools, such as micro grinding pencil tools [4].

However, the presented approach is limited. Cutting forces must be acquired through experimental investigations or by conducting micro cutting simulations. However, care must be taken regarding the implementation of these cutting forces in the presented model:

- Measured cutting forces already include the process-machine-interactions of the employed machine tool and cutting process. Hence, using these forces as an input for the cutting forces to investigate process-machine-interactions is not appropriate.
- Using micro cutting simulations requires precise knowledge of the material parameters of the workpiece and suitable constitutive laws [2]. Further, micro cutting simulations are rather time consuming [25].

Since the process kinematics are modeled using a purely kinematic approach and the machine kinematics only include a simple representation of cutting forces, no elasticity and/or plasticity of the workpiece is considered. Effects, such as ploughing and burr formation, hence cannot be examined.

5 Conclusion and outlook

This paper presented methods for the physical modeling of process-machine-interactions in micro manufacturing. For this purpose, both the process and the machine kinematics were modeled separately and then superimposed. In particular, methods to model air bearing and magnetic bearing spindles were investigated as part of the machine simulation model and micro end milling was investigated as part of the process simulation model.

Machine kinematics were determined by using a fluid dynamics model for the air bearings, an electromagnetic model for the magnetic bearings, and a rotordynamics model for the machine tool spindle rotor. Process kinematics were modeled using a purely kinematic approach. Incorporation of cutting forces was implemented by applying them as an excitation in the rotordynamics model. Finally, process-machine-interactions were modeled by superimposing the machine kinematics and process kinematics under consideration of the kinematic relationship between both components.

The following conclusions can be drawn from this analysis:

- The presented physical model for air bearings is well suited for a broad range of air bearing applications, allowing for new and improved bearing designs while yielding results with higher accuracy compared to existing modeling approaches. Possible applications include the analysis of air bearings with angled injection orifices, the investigation into effects of manufacturing errors on bearing performance, and spherical air bearings;
- Modeling the force-current-displacement relationship $f_x, f_y = f(x, y, i_x, i_y)$ of magnetic bearings with a finite element electromagnetic model and implementing this relationship by means of lookup tables in a control system model provides greater modeling accuracy than using the magnetic circuit method, while keeping computing time acceptable for a moderate number of inputs. This also allows the development of more sophisticated controllers for magnetic bearings. Application possibilities include conducting new analyses of homopolar bearings [17], more aggressive bearing designs in terms of geometry and operating configurations, and spherical magnetic bearings;
- Splitting up the process-machine-interactions simulation model during setup into a physical model of the machine (under consideration of the cutting forces) and a kinematic model of the process provides a feasible method for the simulation-based investigation of process-machine-interactions in micro manufacturing.

Currently, the presented simulations models are being validated to confirm their suitability for the prediction of process-machine-interactions including the surface profile and surface roughness values. In future works, the rotordynamics model will be extended to include a detailed three-dimensional representation of the tool flexibility. Advanced process models will be set up and replace the simple representation of the cutting forces. These process

models may be based on chip formation simulations and can extend the suitability of the physical model to investigate process-machine-interactions in micro end milling by including elasticity and plasticity effects of the workpiece as well as a more accurate representation of the cutting forces. Further, sensitivity studies will be conducted to investigate effects of upstream simulation results on the final tool path and surface roughness results.

References

1 Shukri M. Afazov, Svetan M. Ratchev, Joel Segal, and Atanas A. Popov. Chatter modelling in micro-milling by considering process nonlinearities. *International Journal of Machine Tools and Manufacture*, 56:28–38, 2012. doi:10.1016/j.ijmachtools.2011.12.010.

2 Ravi S. Anand and Karali Patra. Modeling and simulation of mechanical micro-machining—a review. *Machining Science and Technology*, 18(3):323–347, 2014. doi:10.1080/10910344.2014.925377.

3 Jan C. Aurich, Martin Bohley, Ingo G. Reichenbach, and Benjamin Kirsch. Surface quality in micro milling: Influences of spindle and cutting parameters. *CIRP Annals*, 66(1):101–104, 2017. doi:10.1016/j.cirp.2017.04.029.

4 Jan C. Aurich, Marina Carrella, and Michael Walk. Micro grinding with ultra small micro pencil grinding tools using an integrated machine tool. *CIRP Annals*, 64(1):325–328, 2015. doi:10.1016/j.cirp.2015.04.011.

5 George K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge Mathematical Library. Cambridge University Press, 2000. doi:10.1017/CBO9780511800955.

6 Richard Bellman. Perturbation techniques in mathematics, physics and engineering (holt, rinehart and winston, london, 1964), 118 pp., 30s. *Proceedings of the Edinburgh Mathematical Society*, 14(4), 1965. doi:10.1017/S0013091500009068.

7 Christian Brecher, Martin Esser, and Stephan Witt. Interaction of manufacturing process and machine tool. *CIRP Annals*, 58(2):588–607, 2009. doi:10.1016/j.cirp.2009.09.005.

8 Berend Denkena and Ferdinand Hollmann. *Process Machine Interactions*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. doi:10.1007/978-3-642-32448-2.

9 Mohamed E. Eleshaky. Cfd investigation of pressure depressions in aerostatic circular thrust bearings. *Tribology International*, 42(7):1108–1117, 2009. doi:10.1016/j.triboint.2009.03.011.

10 Michael I. Friswell, John E. T. Penny, Seamus D. Garvey, and Arthur W. Lees. *Dynamics of Rotating Machines*. Cambridge Aerospace Series. Cambridge University Press, 2010. doi:10.1017/CBO9780511780509.

11 Qiang Gao, Wanqun Chen, Lihua Lu, Dehong Huo, and Kai Cheng. Aerostatic bearings design and analysis with the application to precision engineering: State-of-the-art and future perspectives. *Tribology International*, 135:1–17, 2019. doi:10.1016/j.triboint.2019.02.020.

12 Siyu Gao, Kai Cheng, Hui Ding, and Hongya Fu. Multiphysics-based design and analysis of the high-speed aerostatic spindle with application to micro-milling. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 230(7):852–871, 2016. doi:10.1177/1350650115619609.

13 Ichiro Inasaki, Bernhard Karpuschewski, and Hwasoo Lee. Grinding chatter–origin and suppression. *CIRP Annals*, 50(2):515–534, 2001. doi:10.1016/S0007-8506(07)62992-8.

14 John D. Jackson. *Classical electrodynamics*. John Wiley & Sons, 3rd edition, 1999.

15 Martin B. G. Jun, Xinyu Liu, Richard E. DeVor, and Shiv G. Kapoor. Investigation of the dynamics of microend milling—part i: model development. *Journal of Manufacturing Science and Engineering*, 128(4):893–900, 2006. doi:10.1115/1.2193546.

16 Sonja Kieren-Ehses, Martin Bohley, Tobias Mayer, Benjamin Kirsch, and Jan C. Aurich. Effect of high spindle speeds on micro end milling of commercially pure titanium. *Proceedings of the 20th euspen International Conference*, pages 61–62, 2020.

17 Eric H. Maslen and Gerhard Schweitzer. *Magnetic Bearings: Theory, Design, and Application to Rotating Machinery*. Springer-Verlag Berlin Heidelberg, 2009. doi:10.1007/978-3-642-00497-1.

18 Christopher Müller, Sebastian Greco, Benjamin Kirsch, and Jan C. Aurich. A finite element analysis of air bearings applied in compact air bearing spindles. *Procedia CIRP - 16th CIRP Conference on Modelling of Machining Operations*, 58:607–612, 2017. doi:10.1016/j.procir.2017.03.337.

19 Christopher Müller, Benjamin Kirsch, and Jan C. Aurich. Compact air bearing spindles for desktop sized machine tools. *Small Machine Tools for Small Workpieces*, 55:21–34, 2017. doi:10.1007/978-3-319-49269-8_2.

20 Tej Pratap, Karali Patra, and Alexander A. Dyakonov. Modeling cutting force in micro-milling of ti-6al-4 v titanium alloy. *Procedia Engineering*, 129:134–139, 2015. doi:10.1016/j.proeng.2015.12.021.

21 Ingo G. Reichenbach. *Beitrag zur Beherrschung der Mikrofräsbearbeitung von Polymethylmethacrylat*. Dissertation, Technische Universität Kaiserslautern, 2017.

22 Pablo Rodríguez and Julio E. Labarga. A new model for the prediction of cutting forces in micro-end-milling operations. *Journal of Materials Processing Technology*, 213(2):261–268, 2013. doi:10.1016/j.jmatprotec.2012.09.009.

23 Dinesh Setti, Peter A Arrabiyeh, Benjamin Kirsch, Marius Heintz, and Jan C Aurich. Analytical and experimental investigations on the mechanisms of surface generation in micro grinding. *International Journal of Machine Tools and Manufacture*, 149:103489, 2020. doi:10.1016/j.ijmachtools.2019.103489.

24 Andras Z. Szeri. *Fluid film lubrication*. Cambridge university press, 2010. doi:10.1017/CBO9780511782022.

25 Thanongsak Thepsonthi and Tuğrul Özel. 3-d finite element process simulation of micro-end milling ti-6al-4v titanium alloy: experimental validations on chip flow and tool wear. *Journal of Materials Processing Technology*, 221:128–145, 2015. doi:10.1016/j.jmatprotec.2015.02.019.

26 Eckart Uhlmann, F Mahr, Y Shi, and U von Wagner. Process machine interactions in micro milling. In *Process Machine Interactions*, pages 265–284. Springer, 2013. doi:10.1007/978-3-642-32448-2_12.

27 Frank Vollertsen, Dirk Biermann, Hans N. Hansen, I. S. Jawahir, and Karl Kuzman. Size effects in manufacturing of metallic components. *CIRP Annals*, 58(2):566–587, 2009. doi:10.1016/j.cirp.2009.09.002.

28 Frank Wardle. *Ultra precision bearings*. Woodhead Publishing, Cambridge, 2015.

29 Tobias Waumans. *On the design of high-speed miniature air bearings: Dynamic stability, optimisation and experimental validation*. Dissertation, Katholike Universiteit Leuven, 2009.

30 David C. Wilcox. *Turbulence modeling for CFD*, volume 2. DCW industries La Canada, CA, 1998.

31 Jens P. Wulfsberg and Adam Sanders. *Small Machine Tools for Small Workpieces: Final Report of the DFG Priority Program 1476*. Lecture Notes in Production Engineering. Springer International Publishing, 2017. doi:10.1007/978-3-319-49269-8.

32 He Chun Yu, Wen Qi Ma, Zu Wen Wang, and Li Fang Xu. Cfd research on aerostatic bearing with tangential supply holes. *Advanced Materials Research*, 97:2021–2026, 2010. doi:10.4028/www.scientific.net/AMR.97-101.2021.

Simulation and Application of a Piezo-Driven System Enabling Vibration-Assisted Micro Milling

Sebastian Greco[1]
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Katja Klauer
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Benjamin Kirsch
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Jan C. Aurich
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Abstract

The ongoing miniaturization of components and the functionalization of surfaces necessitates the improvement of micro machining processes and to increase their efficiency. One method to increase the machining efficiency is reducing the process forces and tool wear, which is achieved by the implementation of vibration-assisted cutting in conventional machining processes. In vibration-assisted cutting, the conventional cutting movement is superimposed by a vibration with defined frequency. By using vibration-assisted cutting technologies, besides increased efficiency, a wider range of materials can be machined. In this paper, vibration-assisted cutting is transferred to micro machining. For this purpose, the design, simulation and application of an easy to integrate system that enables vibration-assisted cutting for micro machining processes is described. The setup was tested using a micro milling process. Two orientations between feed direction and vibration direction were investigated. Frequencies up to 15 kHz were examined, the machined material was brass (CuZn39Pb2). The effect of the superimposed vibration was analysed on the basis of process force, surface roughness, burr formation and slot bottom and was compared with the process results of micro milling without vibration-assistance. A decrease in process forces of up to 63 % was observed during vibration-assisted micro milling.

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases micro machining, micro milling, vibration-assisted cutting, Finite Element Analysis, surface roughness

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.3

Funding The authors would like to thank the German research foundation (DFG) for the financial support within the project AU 185/57-1 "Ultrasonic air bearing spindle for micro machining". Also funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - project number 172116086 - SFB 926.

[1] corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 3; pp. 3:1–3:18

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

The ongoing miniaturization of components and assemblies as well as the functionalization of surfaces used for example in electro mechanical instruments or medical devices [4] lead to a demand for increasing the efficiency and accuracy of micro machining processes. Microstructures, which are essential for such applications, are typically dimensioned significantly smaller than one millimeter. One possibility to produce such filigree structures is the application of micro milling processes with tool diameters smaller 100 µm [1]. Analogous to the machining of larger structures, the accuracy of the micro milling processes is dependent on the machine tool's stiffness [7], arising tool wear and resulting surface quality. In addition, lower tool wear also results in lower process forces [14]. For these reasons and since vibration-assisted machining reduces process forces and tool wear, vibration-assisted machining can be used to significantly increase the accuracy and efficiency of the machining result.
During vibration-assisted machining, the cutting motion of the tool is superimposed by a vibration with a given frequency [10]. The additional relative movement between tool and workpiece leads to changed cutting conditions and reduces process forces as well as tool wear [2]. In addition, the range of machinable materials can be expanded by using vibration-assisted machining [16], which enables applications requesting materials that could not be machined without vibration-assistance. Moreover, vibration-assisted machining is used to produce special surfaces [15], e.g. to improve wettability chracteristics [17]. With regard to micro milling processes with tool diameters of 100 µm and smaller, the use of vibration-assisted machining has not yet been investigated. However, the advantages of the reduced process forces during vibration-assisted machining and the resulting reduced deformation of the machine are promising to further increase the accuracy of the machining result. In order to give an insight into the effects of vibration-assisted machining on the production of structure sizes smaller than 1 mm, a selection of research on vibration-assisted machining with tool diameters of 400 µm and 500 µm is presented below.
In the investigations of Lian et al. [11] the effects of the vibrational amplitude during vibration-assisted micro milling of Al6061 were investigated. A change in the resulting surface roughness in dependence of the vibration's amplitude was observed. The surface roughness reached an optimum at 11 µm vibrational amplitude, which corresponds to a reduction in surface roughness by approx. 52 %. In the investigations, a double edged milling tool with d = 500 µm was used.
In the work of Noma et al. [13] the influence of vibration-assisted machining on the helical milling of glass was investigated. A diamond tool with d = 400 µm, vibrating in axial direction with a frequency of 70 kHz and an amplitude of 4 µm, was used. The investigations showed a significant decrease in process forces by 84 - 86 %. In addition, considerably less tool wear and thus increased tool life as well as a reduced chip size was observed when using vibration-assisted machining.
Supplementing these findings, in this paper the effects of vibration-assistance on micro milling using significantly smaller (d = 100 µm) and single-edged tools is investigated. The design is supported by simulations. The simulative characterization is validated and the final system is applied for the machining of brass (CuZn39Pb2). Different orientations of the feed direction and the direction of the vibration are investigated for frequencies up to 15 kHz. The influence of vibration-assistance is evaluated on the basis of process forces, surface roughness, burr formation and the slot bottom.

2 Design

In this section the requirements and specifications of the system are explained. In addition, the performance characteristics of the system are discussed.

2.1 System Requirements

The system should allow vibration in at least one coordinate direction perpendicular to the tool's axis of rotation (z-axis, see Figure 1). Excitation perpendicular to the z-axis has the advantage that it is not necessary to overcome the weight of the sample for excitation. Even with the small sample sizes in micro machining, large piezo-electric actuators are required when working against the sample's weight. Therefore, in this investigation an excitation perpendicular to the z-axis was used. By adjusting the feed direction during milling, thus the influence of the direction of the vibration on the process result can be examined. Further, the frequency of the vibration needs to be freely adjustable in order to investigate the influence of varying frequencies during vibration-assisted machining on the process results. As the system to enable vibration-assisted machining will be integrated into an already existing ultra-precision machine, the compatibility of the interfaces of both systems is of special importance. As the ultra precision machine contains a dynamometer for the recording of process forces occurring during machining, the dynamometer is defined as the interface between the machine tool and the vibration-assisting system. The interface and a summary of the system requirements are shown in Figure 1.

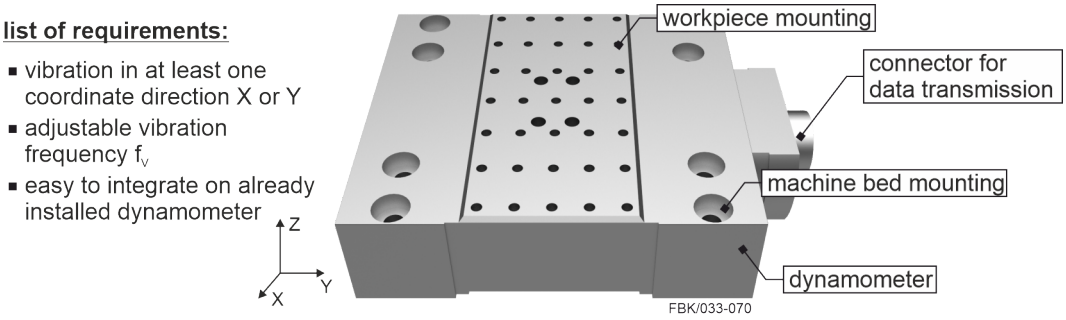


Figure 1 List of requirements and interface for which compatibility must be ensured.

2.2 System Specification

In Figure 2 the designed system to enable vibration-assisted micro milling is shown. The design is similar to the vibration systems used in [17, 9, 3].

Basis of the designed system is a frame, which is mounted directly on the dynamometer and allows an easy integration into already existing systems. On this frame the workpiece holder and the piezo-electric actuator, which generates the vibration, are attached. For the preload of the piezo and the system in general, it has been considered to use as many standard parts as possible. Therefore, the actuator is mechanically preloaded between the frame and the workpiece holder by a standard thread. The actuator's vibration frequency can be controlled via its electric connection. When the piezo-electric actuator changes its length due to an applied voltage, it deflects the workpiece holder. The stiffness of the workpiece holder was reduced in the vibration's direction (x-direction) by using solid-state joints. The modular design of the system enables a quick exchange of the workpiece holder and is not

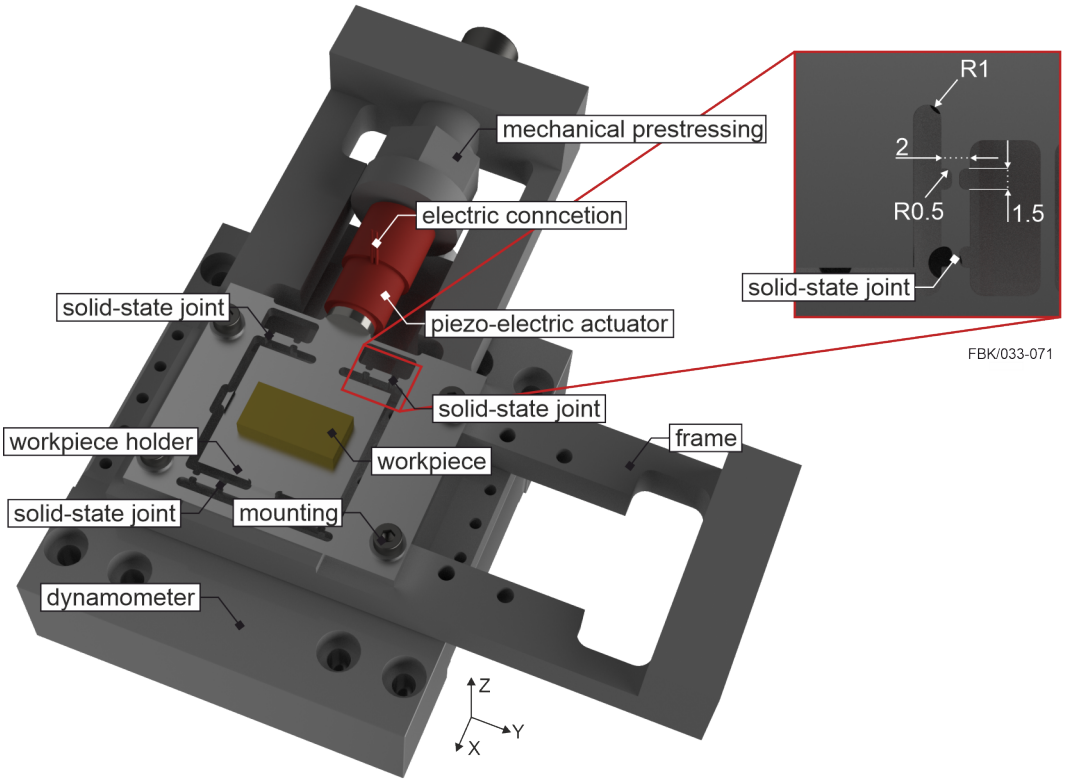


Figure 2 CAD rendered image of piezo-driven system.

bound to a specific geometry of workpiece holders. Furthermore, the system's frame is prepared for the use of a second actuator. The workpiece is attached to the workpiece holder by means of an adhesive joint.

The system is equipped with a piezo-electric ring actuator (Physik Instrumente (PI), P 010.20H PICA1[1]). The actuator is not preloaded at delivery but was mechanically preloaded in the system with 1 MPa due to the increased compressive strength of piezo-electric actuators compared to their tensile strength. The maximum travel Δl_0 of the actuator is 30 µm at an operating voltage of 1000 V. The sinusoidal electric signal to control the actuator is generated by a frequency generator (Hameg, HM 8030 31[1]) and is amplified by a controller (Physik Instrumente (PI), E 618.10G1[1], max. frequency: 15 kHz). With the system's stiffness in x-direction of 2.23 N/µm (calculated in section 3.1), this setup allows a theoretical vibrational amplitude of 1.8 µm at frequencies up to 15 kHz. Due to the low feed per tooth in micro milling of $f_z = 1$ µm and the spindle's speed of n = 24,420 min^{-1} (see section 4), the system ensures that the tool exits the material during machining and is therefore suitable for the application in vibration-assisted micro machining.

3 Finite Element Analysis

In the following the simulative characterization of the system is described by means of a convergence study for the stiffness analysis and a modal analysis. The software ANSYS Workbench 19.211 was used to run the simulations.

3.1 Stiffness Analysis

In order to calculate the maximum displacement of the piezo-electric actuator, the system's stiffness has to be known. In general, simulation results become more accurate as the mesh becomes finer. However, due to the increasing complexity of the mesh, the fineness of the mesh is directly linked to the calculation time required for the simulation. In order to determine the element size at which the simulation result is no longer subject to major changes, several simulations with varying mesh sizes need to be compared. Therefore, a study to determine the mesh's convergence was carried out. For this purpose, the stiffness of the designed system in the vibration's direction (x-direction) was calculated using Finite Element Analysis at different mesh sizes (0.4 - 4 mm). The convergence criterion was defined as a change in the simulation result of less than 3 % within three successive mesh refinements. The boundary conditions for the calculation of the static stiffness and an exemplary meshing of the workpiece holder and the workpiece with an element size of 4 mm are shown in Figure 3.

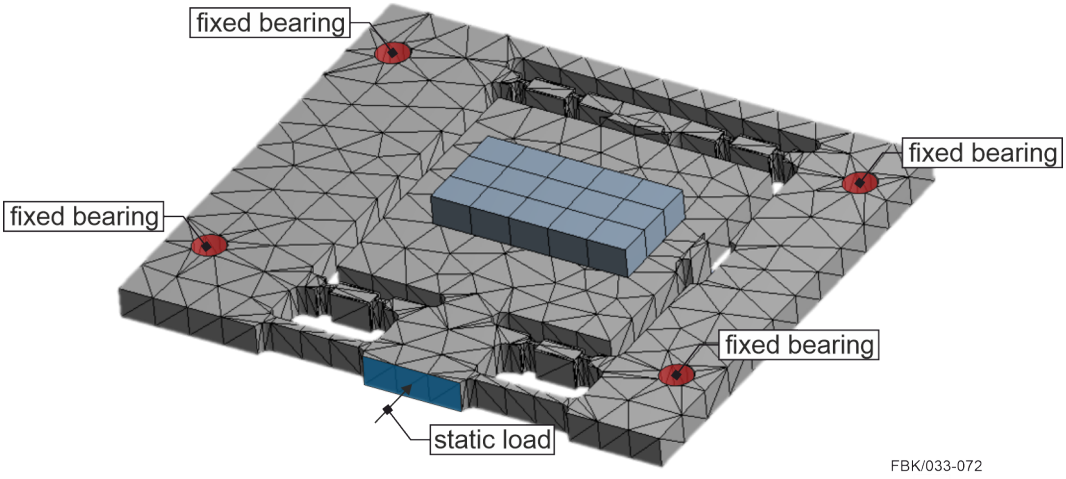


Figure 3 Boundary conditions for Finite Element Analysis and exemplary representation of the mesh with an element size of 4 mm.

The mounting holes for fixing the workpiece holder onto the frame have been defined as fixed bearings, which suppresses their movement for all degrees of freedom. As the correlation between force and deflection is linear, a static load of 10 N has been chosen as force acting on the contact surface between the piezo-electric actuator and the workpiece holder. This force leads to small deflections of the system in the single-digit micrometer range, which is similar to the system's application and thus leads to comparable conditions between simulation and reality. The stiffness of the workpiece holder was calculated by the ratio of the given force and the simulated displacement in direction of the applied force. In Figure 4 the results of the Finite Element Analsysis are given, showing the deflected workpiece holder for an element size of 4 mm. The color scale indicates the deflection in x-direction, that is, in the direction of the vibration generated by the piezo-electric actuator. The uniform coloring of the inner part of the workpiece holder shows that the kinematics of the workpiece holder allows an equal movement of the workpiece in x-direction.

In Table 1 the results of the convergence study are shown. For each simulation the element size, the number of nodes and elements of the mesh, as well as the deflection and stiffnes are shown. In addition, the percentage change in stiffness to the next larger mesh is given.

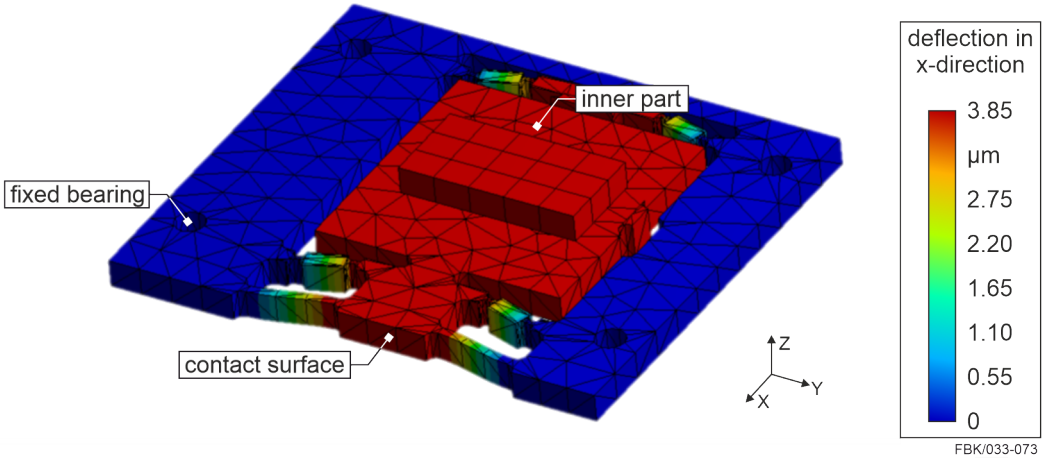


Figure 4 Deflected system shown for an element size of 4 mm.

Table 1 Simulation results of the convergence study calculating the workpiece holder's stiffness.

element size in mm	number of nodes	number of elements	deflection in µm	stiffness in N/µm	percentage change
4.0	5,800	2,630	3.83	2.61	-
3.0	8,290	3,969	3.85	2.60	-0.46 %
2.0	18,123	9,523	3.87	2.58	-0.68 %
1.0	99,538	61,806	4.18	2.39	-7.30 %
0.8	185,298	118,150	4.28	2.34	-2.36 %
0.6	411,591	271,224	4.38	2.28	-2.28 %
0.4	1,325,754	903,466	4.49	2.23	-2.45 %

The results of the mesh study show stiffness values from 2.61 - 2.23 N/µm with varying element sizes from 4 - 0.4 mm. Considering the percentage change of the stiffness, the biggest change (-7.30 %) is observed when refining the mesh's element size from 2 to 1 mm. Starting at an element size of 0.8 mm and below, however, only slight changes within the lower single digit percentage range were found in the simulation results, which complies the stop criterion and indicates that the mesh is sufficiently accurate to use the stiffness for the calculation of the actuator's displacement.

According to [12], the actuator's travel for alternating external loads Δl is given by:

$$\triangle l = \frac{\triangle l_0 \cdot c_{\mathrm{System}} - F}{c_{\mathrm{System}} + c_{\mathrm{Piezo}}} \tag{1}$$

with the actuator's maximum travel Δl_0 and the constant force of mechanical preload F. The system's stiffness c_{System} of 2.23 N/µm and the mechanical preload F are relatively low compared to the stiffness of the piezo-electric actuator (manufacturer's information: $c_{\mathrm{piezo}} = 68$ N/µm), which leads to a minor influence of the actuator's travel. Due to the controller's output voltage of 130 V and the piezo's linear operating range [8] a travel of 3.6 µm is reached, leading to a vibrational amplitude of 1.8 µm. Due to the stiffness of the solid-state joints in the direction of the vibration (x-direction), the actuator's travel and therefore the vibration's amplitude is suitable to enable vibration-assisted machining.

3.2 Modal Analysis

The travel of the piezo-electric actuator is represented by a vibration in x-direction which is transferred to the workpiece. When the actuator's vibrational frequency matches the natural frequencies, unwanted movements of the workpiece can occur. However, it is also possible to further amplify the intended movement. To anlyze this, a simulative modal analysis was carried out.

In Figure 5 the simulation results of the modal analysis are listed, which show the system's natural frequencies up to 15 kHz. Two different examples of modes that occur due to vibration in natural frequencies are illustrated. The shown deformation of the workpiece holder when vibrating in the natural frequency of 1959 Hz is identical with the movement induced by the piezo-electric actuator. Due to the positive interference between excitation vibration and natural frequency, the desired motion of the workpiece holder is enhanced which leads to an increase of the vibration's amplitude when excited with 1959 Hz and harmonics of this natural frequency. The natural frequency at 1959 Hz is the only vibration up to 15 kHz at which the modal analysis revealed an amplification of the vibration. All other natural frequencies have to be avoided.

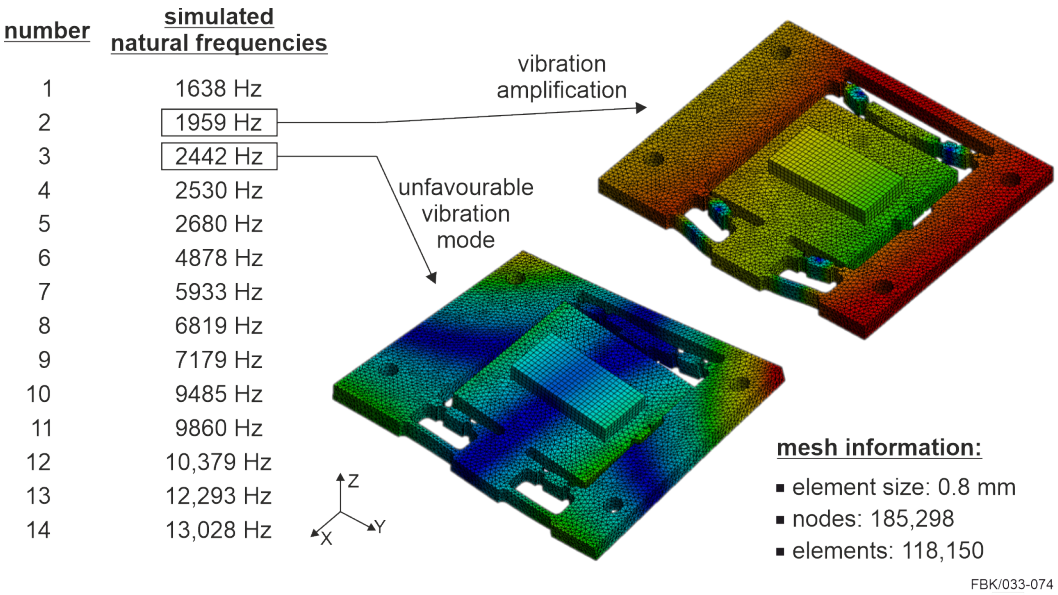


Figure 5 Simulated natural frequencies up to 15 kHz and representation of different vibration modes.

The negative influence of natural frequencies other than 1959 Hz on the system's motion is also demonstrated in Figure 5, where the mode of the natural frequency vibration at 2442 Hz is shown. It can be seen that the equal displacement in x-direction of the inner part of the workpiece holder and the workpiece itself is superimposed by numerous undesired displacements. The most significant motion in this example is a rotation around the x-axis, which causes the workpiece surface to move along the z-axis. Since the amplitudes of natural frequency vibrations are generally more intense than those of forced vibrations, it can be assumed that at this frequency only the disruptive displacement can be observed. These disruptive movements significantly disturb the accuracy of micro machining and can lead to tool breakage. The results of the modal analysis were used for the design of the experiment in order to avoid natural frequencies and to only meet the natural frequency of 1959 Hz and its harmonics at which the intended motion of the workpiece is favourably enhanced.

4 Experimental Setup

The experimental tests were performed on a LT Ultra MMC 600H[1] machine tool. This 5-axes ultra-precision milling machine features hydrostatic mounted axes. The x-, y- and z-axis are driven by linear motors, the rotary axes are driven by torque motors. The maximum rotational speed of the air bearing spindle (Levicron ASD-80-H25[1]) is 80,000 rpm. Brass CuZn39Pb3 was used as workpiece material. The workpiece (20 x 10 x 3 mm) was glued to the workpiece holder, which was mounted on the system's frame (see section 2.2). The vibration system was mounted on the Kistler 9119AA2[1] dynamometer, which was fixed to the machine tool. The workpiece was face milled in the identical clamping used for the tests prior to the vibration-assisted experiments. All experiments were carried out using one workpiece and clamping in order to avoid changes of the system's natural frequencies due to smallest variations of the piezo's mechanical preload or the workpiece's mass and position during re-clamping. The experimental setup is shown in Figure 6a.

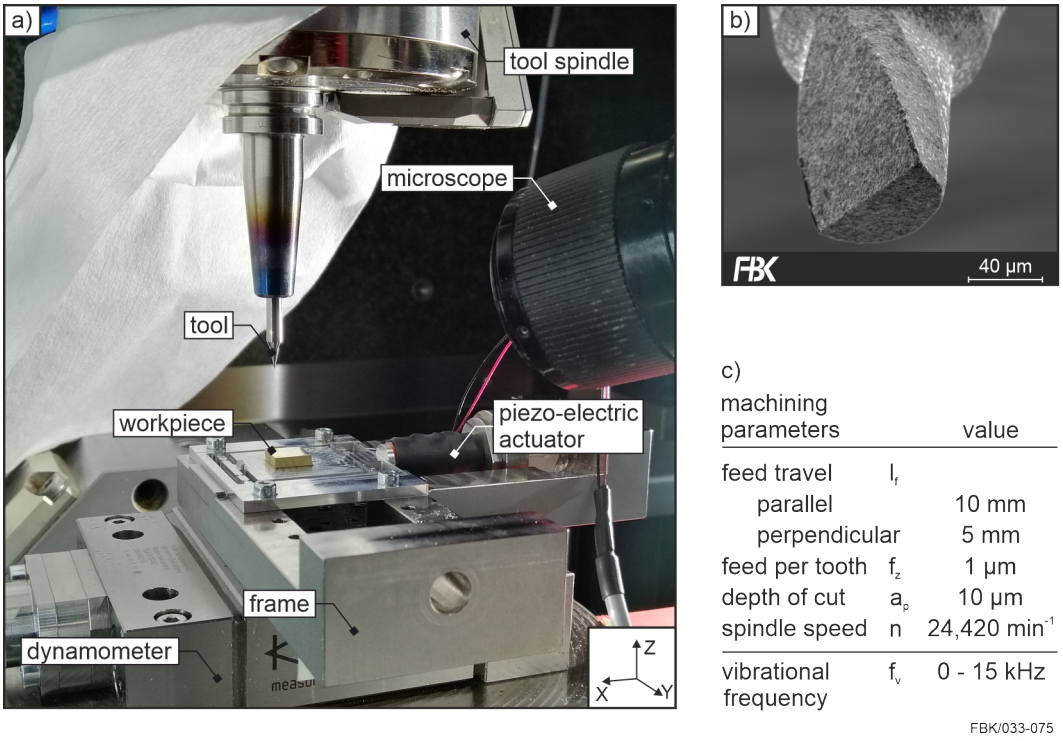


Figure 6 a) Experimental setup, b) micro milling tool, c) cutting parameters.

A single-edged cBN-tool (NS tools; SMEZ120 D0,100[1]) with an effective radius of 50 µm was used for machining (see Figure 6b). This tool was clamped in a Levicron HSW-E[1] shrink chuck: the static concentricity error is below 0.8 µm (manufacturer's specification). All experiments were carried out without metalworking fluid. The process was monitored with a microscope camera. When choosing the spindle speed, it must be ensured that it does not coincide with the frequencies to be tested. If the two frequencies were equal, they could not be distinguished from each other in the force signal within the frequency analysis (Fourier transformation of the force signal, see section 5.1). Thefore, the spindle speed was set to 24,420 rpm, which corresponds to 407 Hz and does not coincide with any of the excitation frequencies (see Table 2). Due to the superposition of the tool's cutting movement and the

workpiece's vibration, the number of oscillations between tool and workpiece per rotation m is given by the ratio of the vibration's frequency to the spindle's rotational frequency:

$$m = \frac{f_v \cdot 60}{n} \tag{2}$$

The frequencies of the vibrations examined were defined as follows: The process forces were monitored in real time. Only the offset (tool not engaged) of the force was recorded. The sample was vibrated with the described setup and operated at frequencies of 0 - 15 kHz. The frequency at which the force offset became maximum was detected. At this frequency, a natural frequency or a harmonic of a natural frequency is to be assumed, which causes the workpiece holder to vibrate. The frequency detected with this method, generating the largest vibration, was about 3.8 - 3.9 kHz. This experimentally determined frequency corresponds to twice the natural frequency (1st harmonic) at which the simulation showed an amplification of the desired motion by the vibration. At the natural frequency an amplification of the oscillation was also noticed, but the intensity was lower than at the first harmonic. The frequency area around the resonance frequency (3.8 - 3.9 kHz) was investigated more closely in the subsequent test, the tests without excitation and with excitation at the resonance frequency were repeated three times for statistical verification (see Table 2).

Table 2 Tested frequencies of vibration-assisted micro milling.

Tested frequency of vibration in kHz	Number of oscillations per spindle rotation m	Repetitions for statistical verification
0	0	3
1	2.5	1
2	4.9	1
3	7.4	1
3.6	8.8	1
3.7	9.1	1
3.8	9.3	3
3.9	9.6	3
4.0	9.8	1
5.0	12.3	1
6.0	14.7	1
10.0	24.6	1
15.0	36.9	1

The maximum sampling rate of the force measurement using the dynamometer was 12 kHz. Due to the fact that the sampling rate according to the Nyquist theorem must be more than twice the vibration's frequency, a frequency analysis of the measured process forces is only possible up to 6 kHz. Accordingly, only two frequencies above 6 kHz were tested.

During the test, slots were milled either parallel or perpendicular to the direction of vibration (see Figure 7). The vibration was carried out in the same direction in all experiments, according to the design of the workpiece holder described in section 2.2. Therefore, the feed direction was adjusted accordingly to realize both cases (parallel and perpendicular to the vibration's direction). The feed travels for the slots are listed in Figure 6c. The different lengths of feed travels were chosen to avoid the crossing of slots milled with different feed direction.

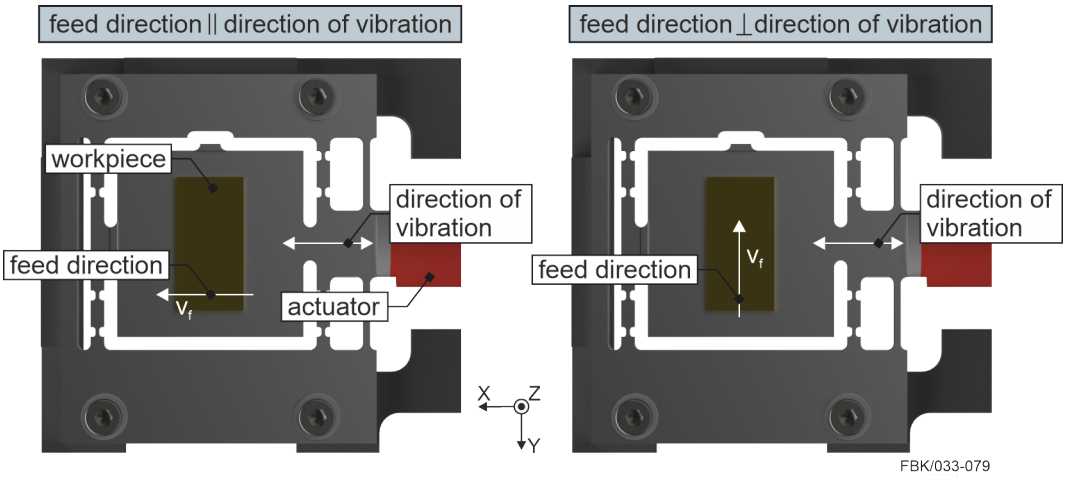


Figure 7 Variation of the feed direction in relation to the direction of vibration.

After machining, all slots were characterized with a confocal microscope (Nanofocus µSurf explorer[1]). The objective used features a 50x magnification and a numerical aperture of 0.5. In order to determine the areal surface texture parameters, three images were taken per slot: one image at the beginning of the slot, one in the middle of the slot and one image at the end of the slot. The measuring field of the three images corresponded to a single field size of the objective used (320 µm x 320 µm). In addition to the areal analysis, a profile analysis was also carried out. For this purpose, a larger measuring field, which was stitched from two single fields, was measured in the middle of each slot. One profile section in the middle of the slots was taken from this larger measuring field for the profile evaluation. The processing of the data was carried out in MountainsMap[1] as follows for the areal and profile evaluation:

Areal surface texture parameters

- Untilting of the measured data (fitting of a plane)
- Extraction of the evaluation area: size: 75 µm (width) x 100 µm (length). Not the entire width of the slot was extracted since burr partially protruded into the slot. In order to eliminate this area for evaluation, only a width of 75 µm was used. This central area was extracted in the same way for all slots.
- Filtering: ls = 0.25 µm
- Calculation of the surface texture parameter Sa according to ISO 25178-2 [6]

Profile roughness parameters

- 480 µm long profile is extracted in the middle of the slot
- Filtering: ls = 0.25 µm and lc = 8 µm
- Calculation of the roughness parameter Ra according to ISO 4287 [5]

In order to qualitatively examine the resulting topographies at the slot bottoms and the burr formation, scanning electron microscope (SEM) images were taken in addition to the described measurements with the confocal microscope.

During machining, the process forces were detected with a dynamometer (Kistler 9119AA2[1]). The recorded process forces were then evaluated in Matlab[1]. First, long-wavelength components, which can be caused by drifts, were filtered out. A high-pass

filter with a cut-off frequency of 1 Hz was used. Then the offset of the recorded forces was determined. This offset contained the inertial forces resulting from the vibration of the inner part of the workpiece holder and the workpiece itself when the tool is not in engagement. The offset was subtracted from the total force to determine the force acting on the tool. The evaluated resultant force value is derived from the mean length value of the vector addition of the measured forces in x-, y- and z-direction. The specific resultant forces were calculated by determining the individual slot depths and the ratio between force to individual slot depth.

5 Experimental Results

The simulative characterization of the system via the modal analysis (section 3.2) was validated experimentally. Subsequently, vibration-assisted micro milling was done for several vibrational frequencies with feed direction parallel and perpendicular to the direction of vibration. The experimental results were evaluated on basis of the process forces, the surface roughness, the burr formation and the slot bottom quality.

5.1 Vibration Analysis

In Figure 8 a linear interpolation of the results of the Fourier transformation (FFT) of the force signal in the direction of the vibration (x-direction) is shown for vibration frequencies between 0 - 6 kHz.

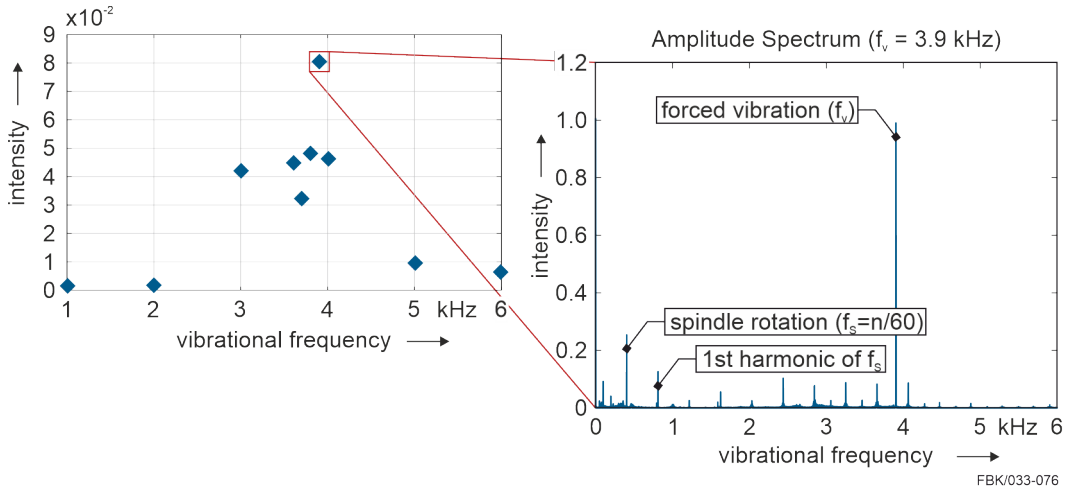


Figure 8 Measured system response to forced vibration and amplitude spectrum for $f_v = 3.9$ kHz.

The average deviation between the frequency set at the frequency generator and the measured frequency (x-coordinate of the forced vibration in the FFT, see Figure 8) was 0.2 %. The intensity of the vibrations determined during tool engagement show a clear peak at $f_v = 3.9$ kHz. As described in section 4, a significant increase of the inertial forces in the direction of the vibration was observed at this frequency. This resonant frequency can be assumed to be the 1st harmonic of the systems natural frequency at 1959 Hz (see section 3.2). Since the real time monitoring of forces showed that the resonance at $f_v = 3.9$ kHz was more intense than the resonance at the natural frequency, vibration-assistance at this resonance frequency causes the strongest amplification of the system's vibrational amplitude. In addition to the intensities of the vibrations generated by the piezo-electric actuator, the detailled

amplitude spectrum at the vibrational frequency of $f_v = 3.9$ kHz is also given in Figure 8. The analysis of the spectrum shows two vibrations that are particularly distinctive. One of these two vibrations is due to the spindle's rotation at $f_S = n/60 = 407$ Hz. The peaks that repeat at regular intervals are harmonics of f_S and can also be attributed to the vibration caused by the spindle's rotation. The most distinctive vibration in the shown amplitude spectrum is the vibration at f_v which is generated by the piezo-electric actuator. The comparison of the intensities shows that the vibration at f_v clearly exceeds that of the spindle's vibration. The high intensity is due to f_v meeting the 1st harmonic of the workpiece holder's natural frequency at 1959 Hz, which leads to a resonance of the system and an amplification of the mode of motion of this natural frequency. The actual resonance frequency deviates by less than 1 % from results of the modal analysis. The results of the modal analysis are therefore considered validated. As this mode of motion equals the workpiece holder's motion generated by the piezo-electric actuator, it can be assumed that the amplitude during vibration-assisted micro milling is significantly increased at $f_v = 3.9$ kHz.

5.2 Process Forces

In Figure 9a the specific resultant forces are shown for the feed direction parallel and perpendicular to the direction of vibration and for varying vibrational frequencies f_v. In general, the profiles of the forces are very similar. However, slightly increased forces are observed when the feed direction is perpendicular to the direction of vibration. During micro milling specific resultant forces between 0.0016 - 0.0044 N/µm for feed direction parallel to the direction of vibration and 0.0026 - 0.0056 N/µm for feed direction perpendicular to the direction of vibration were detected.

During micro milling without vibration ($f_v = 0$ Hz) no dependence of the specific resultant forces on the feed direction was observed which is due to the non-existent vibration and the therefore identical process independent from the feed direction. The slightly increased forces when micro milling with $f_v > 0$ Hz and feed direction perpendicular to the vibration's direction is due to the orientation of these directions. When micro milling with feed direction

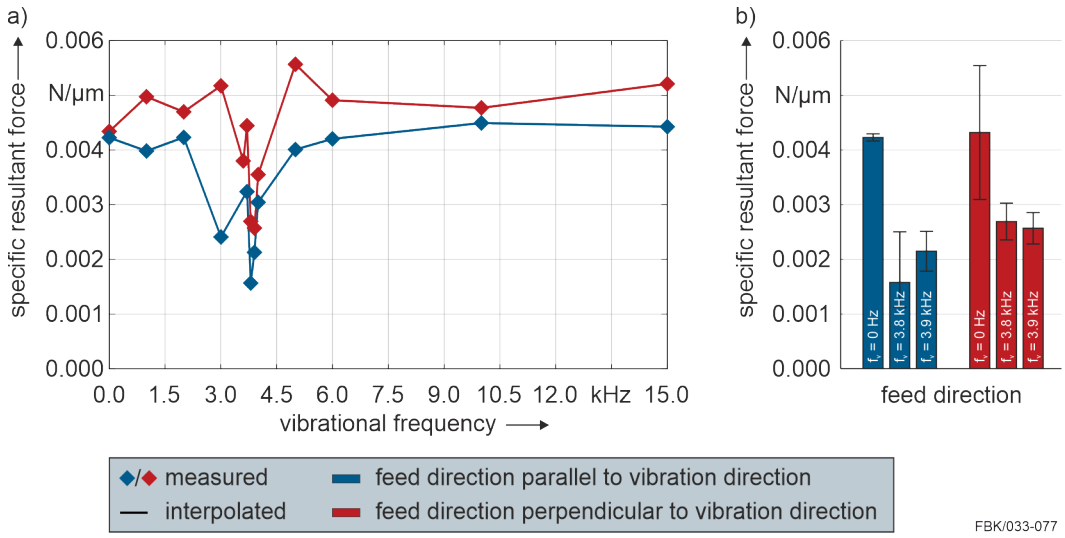


Figure 9 a) Specific resultant forces as function of vibrational frequency, b) comparison resultant force without vibration-assistance and at natural frequency for both feed directions.

perpendicular to the direction of vibration, the vibration leads to an increase of the micro milled slot's width, which results in an increased material removal and therefore increased forces. During micro milling with feed direction perpendicular to the direction of vibration and $f_v = 3.9$ kHz, an increase of the slot's width by approx. 2 µm compared to $f_v = 0$ Hz was observed. However, the measurement is interfered by burrs protruding into the slot, which can lead to deviations of the determined absolute value and the calculated amplitude of the vibration (see section 3.1). In Figure 9b the specific resultant forces and standard deviations for micro milling without vibration-assistance and with vibration in the range of the 1st harmonic of the workpiece holder's natural frequency are shown for both feed directions.

A reduction of the resultant forces during vibration-assisted micro milling is confirmed for both feed directions. Micro milling with $f_v = 0$ Hz and feed direction perpendicular to the direction of vibration shows large variations of the forces compared to the identical process with feed direction parallel to the direction intended for vibration-assistance. These differences in resultant forces are attributed to the workpiece holder's comparatively low stiffness in x-direction due to the solid-state joints. This reduced stiffness causes that the solid-state joints of the workpiece holder allow minimal movements of the workpiece perpendicular to the feed direction due to the process forces during micro milling without vibration. During vibration-assisted micro milling with vibrational frequencies $f_v > 0$ Hz the solid-state joint's displacement is defined by the vibration which avoids such undesired movements. As the feed of the tool preloads the solid-state joints of the workpiece holder when micro milling with $f_v = 0$ Hz and feed direction parallel to the intended direction of vibration, the effect of reduced stiffness during micro milling without vibration cannot be observed, which is also expressed in the significantly lower standard deviations.
Both feed directions show a strong decrease in resultant forces in the range of 3.8 - 3.9 kHz, in which the 1st harmonic of the workpiece holder's natural frequency of 1959 Hz is located. Comparing vibration-assisted micro milling at this resonant frequency to micro milling without vibration ($f_v = 0$ Hz), resultant forces were reduced by 63 % for the feed direction parallel to the direction of vibration and 41 % for the feed direction perpendicular to the direction of vibration. The lower level of resultant forces during micro milling with vibration-assistance is also given in Figure 10, showing the moving average of 1200 values of the resultant force over the corresponding process time. The maximum resultant forces for micro milling with vibration-assistance are significantly lower (0.046 N) compared to micro milling without vibration-assistance (0.070 N), which reduces the risk of tool breakage due to short-term high loads during machining.

This reduction of the forces is assumed to be due to the resonance of the workpiece holder in this frequency range, which leads to a significantly increased vibrational amplitude and thus higher velocity of the workpiece during vibration. If the speed due to the vibration exceeds the feed speed, the machining is governed by an interrupted cut [2]. We believe that this strong increase in vibrational amplitude leads to a reduced time of tool engagement which reduces the process forces. The relatively lower decrease in process force during micro milling with feed direction perpendicular to the direction of vibration compared to micro milling with feed direction parallel to the direction of vibration is due to the superposition of the increased slot width at $f_v = 3.9$ kHz compared to $f_v = 0$ Hz. As the resonant frequency of the system is no longer excited with increasing vibrational frequency and hence the amplitude of the workpiece's vibration decreases again, the cutting forces increase for $f_v > 3.9$ kHz.

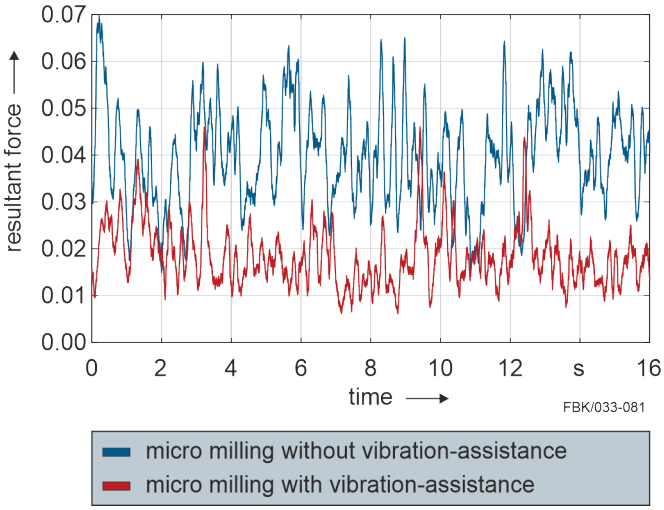


Figure 10 resultant forces with vibration-assistance (f_v = 3.9 kHz) and without vibration-assistance.

5.3 Surface Roughness

Figure 11a shows the trend of the areal surface texture parameter Sa for the different vibrational frequencies. The measured data points in the diagram represent the mean values based on the three individual measurements (beginning, middle and end of the slots). The given average standard deviation $\bar{\sigma}$ are the mean values of the standard deviation of all measurements for the respective feed direction. In Figure 11b the values of the profile roughness parameter Ra for selected vibrational frequencies are given. Here, only slots which were manufactured without vibration and slots which were manufactured with vibration at the resonance frequency are depicted. For each of these two special cases three slots were manufactured (see section 4). Each of the three slots was measured and evaluated as described, so that for each bar shown in Figure 11b nine measured values were considered (three manufactured slots, each measured at three positions). The scattering between the individual slots was displayed as error bars.

On basis of the integral parameter Sa, no trend with regard to the surface roughness as a function of the vibrational frequency can be identified when the feed direction is perpendicular to the vibration direction. The behavior is different when the feed direction is oriented parallel to the direction of the vibration. It can be figured out that the value of Sa decreases with increasing vibrational frequency and then remains at a constant level from a frequency of 6 kHz. In the range of the resonant frequency, a local maximum can be detected. However, at this point the roughness is still below the roughness for the slots that were milled without vibration-assistance. The varying roughness depending on the feed direction could be due to cutting mechanisms that occur at the slot's bottoms during machining. When the feed direction is perpendicular to the direction of vibration, the slot is wider, however, the vibration seems to have no significant influence on the kinematic roughness at the slot's bottom. Based on these observations, we believe that vibration-assisted machining with the direction of the vibration being parallel to the feed direction improves the surface quality by increasing the distance between two adjacent milling tracks and thus reducing the kinematic roughness. In addition, the feed rate is varied by the vibration which could lead to a reduction of the area at the edge of the micro milled slots where the minimum chip thickness is not reached anymore.

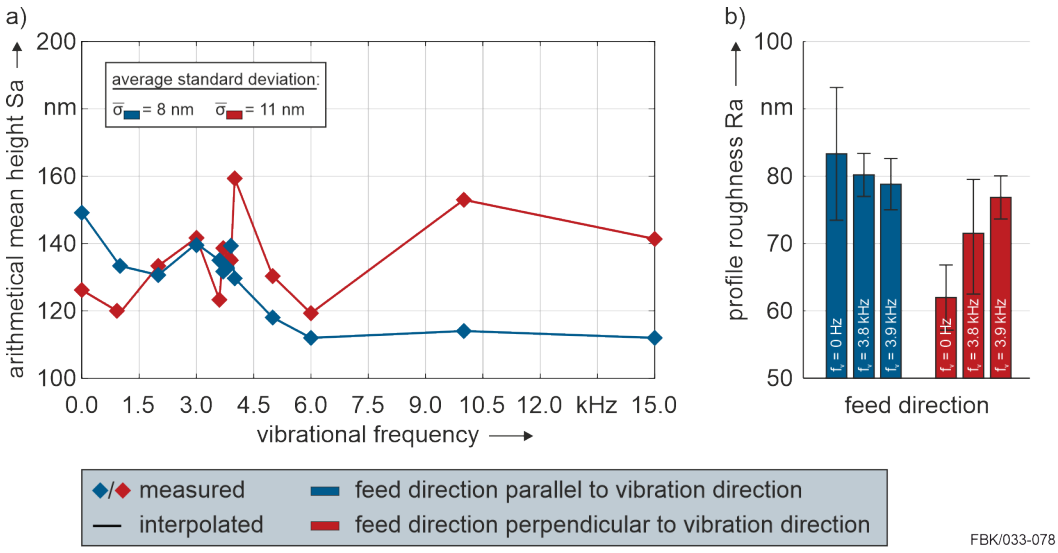


Figure 11 a) Arithmetical mean height Sa as function of vibrational frequency, b) comparison of profile roughness Ra without vibration-assistance and at resonance frequency for both feed directions.

However, the reduction of roughness does not apply to the frequency range of resonance, which is why we assume that there are overlaying effects caused by the significant increase of the vibration's amplitude. For the slots which were manufactured with feed direction parallel to the vibration's direction, a reduction of roughness when micro milling with f_v in the range of the workpiece holder's resonance can also be seen for the profile roughness parameter Ra. The values for Ra are slightly smaller in the case of vibration-assistance. The less significant reduction of Ra compared to the area based Sa may be due to the sole consideration of the profile in the middle of the slot and the neglection of areas near the slot's edge, highlighting the influence of the minimum chip thickness at the sides of the slot. It is also noticeable that most of the error bars for the slots, which were manufactured with vibration, are significantly smaller than those which were manufactuered without vibration. This means that the three successively manufactured slots exhibit only small deviations from each other in terms of their profile roughness. Vibration-assistance parallel to the feed direction thus apparently also has a positive effect on the repeatability with regard to integral roughness parameters.

5.4 Burr Formation and Slot Bottom

Figure 12 shows the burr formation and the surface of the slot bottoms for manufacturing without vibration-assistance, manufacturing with vibration at the resonant frequency and for manufacturing at the highest frequency investigated in this research. Two SEM images are shown for each slot: the image with the smaller magnification is suitable for assessing the burr formation, the image with the higher magnification shows the slot bottoms.

Solely the images in the first row (no vibration-assistance, $f_v = 0$ Hz) show slight differences in burr formation. These differences could be due to the reduced stiffness of the workpiece holder perpendicular to the feed direction (y-direction), similar to the scattering of the process forces during micro milling with feed direction perpendicular to the direction of the intended vibration. For $f_v > 0$ Hz, neither with regard to burr formation nor to the surface quality at the slot bottoms, the SEM images show any trend in dependence on the feed direction or vibrational frequency. For vibration-assisted micro milling with feed direction

parallel to the direction of vibration, no differences in burr formation and slot bottoms were found compared to $f_v = 0$ Hz. For $f_v > 0$ Hz, the workpiece holder's movement is defined by the vibration and no differences due to the feed direction could be detected. Based on these results it can be assumed that vibration-assistance has no influence on the burr formation during micro milling. Since the quantitative differences in surface roughness are in the low two-digit nanometer range (see section 5.3), there are no significant differences with regard to the surface quality in the slot bottoms when comparing the SEM images qualitatively.

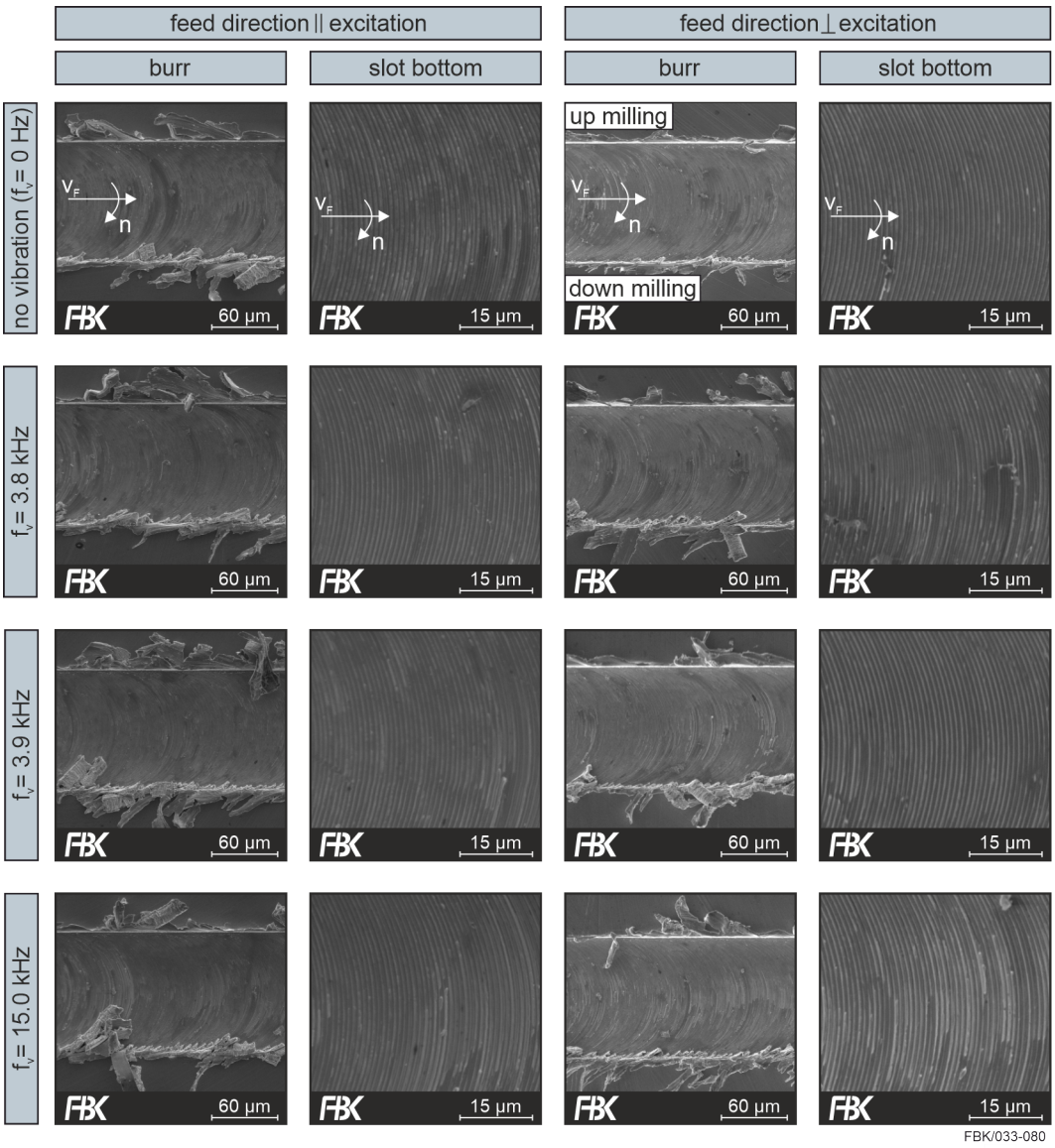


Figure 12 SEM-images of the micro milled slots showing burr and slot bottom for varying feed directions and excitation frequencies.

6 Conclusion

The research presented in this paper focused on the simulation and application of a piezo-driven system to enable vibration-assisted micro milling. In order to calculate the maximum amplitude of the vibration, the stiffness of the system in direction of the vibration was determined using Finite Element Analysis. Furthermore, the natural frequencies of the system were determined by a simulative modal analysis and vibration amplifying frequencies were identified. The system was applied for vibration-assisted machining of brass (CuZn39Pb2) with vibrational frequencies from 0 - 15 kHz. The simulative determination of the system's natural frequencies was validated by a Fourier transformation (FFT) of the force signal. The influence of vibration-assistance on micro milling was evaluated on the basis of process forces, surface roughness, burr formation and the slot bottom quality. Based on the experimental results, the following conclusions were drawn:

- Vibration in the resonant range of the system leads to a strong increase in the vibrational amplitude and a significant reduction in the process forces due to a reduced time of tool engagement.
- The orientation of the feed direction to the direction of the vibration influences the surface roughness. When these run parallel, the roughness at the slot's bottom is reduced; when these are oriented perpendicular to each other, no significant influence on the roughness was observed.
- With regard to the slot bottom appearance and burr formation evaluated via SEM-images, no difference was found between vibration-assisted micro milling and micro milling without superimposed vibration.

In future investigations the system will be extended by a second piezo-electric actuator in order to enable the superposition of two individual vibrations and to investigate the influence of bidirectional vibration-assistance on micro milling. In addition, since the reduction of process forces is the most pronounced effect of vibration-assistance observed in this investigation and leads to lower tool wear, tool wear during micro milling with increased material removal rates using vibration-assistance will be studied.

[1]Naming of specific manufacturers is done solely for the sake of completeness and does not necessarily imply an endorsement of the named companies nor that the products are necessarily the best for the purpose.

References

1 J.C. Aurich, I.G. Reichenbach, and G.M. Schüler. Manufacture and application of ultra-small micro end mills. *CIRP Annals - Manufacturing Technology*, 61(1):83–86, 2012.

2 D.E. Brehl and T.A. Dow. Review of vibration-assisted machining. *Precision Engineering*, 32:153–172, 2008.

3 W. Chen, L. Zheng, X. Wenkun, K. Yang, and D. Huo. Modelling and experimental investigation on textured surface generation in vibration-assisted micro-milling. *Journal of Materials Processing Technology*, 266:339–350, 2019.

4 D. Dornfeld, S. Mina, and Y. Takeuchi. Recent advances in mechanical micromachining. *CIRP Annals - Manufacturing Technology*, 55(2):745–768, 2006.

5 DIN e.V. *DIN EN ISO 4287:2010-07, Geometrische Produktspezifikationen (GPS) – Oberflächenbeschaffenheit: Tastschnittverfahren – Benennungen, Definitionen und Kenngrößen der Oberflächenbeschaffenheit.* Beuth Verlag, 2010.

6 DIN e.V. *DIN EN ISO 25178-2, Geometrische Produktspezifikationen (GPS) – Oberflächenbeschaffenheit: Flächenhaft – Teil 2: Begriffe und Oberflächen-Kenngrößen.* Beuth Verlag, 2012.

7 X. Gao, B. Li, J. Hong, and J. Guo. Stiffness modeling of machine tools based on machining space analysis. *International Journal of Advanced Manufacturing Technology*, 86(5-8):2093–2106, 2016.

8 K. Janschek. *Systementwurf mechatronischer Systeme.* Springer-Verlag, 2010.

9 X. Jin and B. Xie. Experimental study on surface generation in vibration-assisted micro-milling of glass. *International Journal of Advanced Manufacturing Technology*, 81:507–512, 2015.

10 M.N. Kuma, S.K. Subbu, P.V. Krishna, and A. Venugopal. Vibration assisted conventional and advanced machining: A review. *Procedia Engineering*, 97:1577–1586, 2014.

11 H. Lian, Z. Guo, Z. Huang, Y. Tang, and J. Song. Experimental research of al6061 on ultrasonic vibration assisted micro-milling. *Procedia CIRP*, 6:561–564, 2013.

12 T. Melz. *Entwicklung und Qualifikation modularer Satelitten-Systeme zur adaptiven Vibrationskompensation an mechanischen Kryokühlern.* Technische Universität Darmstadt, 2001.

13 K. Noma, Y. Takeda, T. Aoyama, Y. Kakinuma, and S. Hamada. High-precision and high-efficiency micromachining of chemically strengthened glass using ultrasonic vibration. *Procedia CIRP*, 14:389–394, 2014.

14 I. G. Reichenbach, M. Bohley, F. J. Sousa, and J. C. Aurich. Tool-life criteria and wear behavior of single-edge ultra-small micro end mills. *Precision Engineering*, 55:48–58, 2019.

15 D. Xing, J. Zhang, X. Shen, Y. Zhao, and T. Wang. Tribological properties of ultrasonic vibration assisted milling aluminium alloy surfaces. *Procedia CIRP*, 6:539–544, 2013.

16 L. Zheng, W. Chen, and D. Huo. Review of vibration devices for vibration-assisted machining. *International Journal of Advanced Manufacturing Technology*, 108(5-6):1631–1651, 2020.

17 L. Zheng, W. Chen, M. Pozzi, X. Teng, and D. Huo. Modulation of surface wettability by vibration assisted milling. *Precision Engineering*, 55:179–188, 2019.

Visitation Graphs: Interactive Ensemble Visualization with Visitation Maps

Anna-Pia Lohfink
Technische Universität Kaiserlautern, Germany

Christoph Garth
Technische Universität Kaiserlautern, Germany

Abstract

Modern applications in computational science are increasingly focusing on understanding uncertainty in models and parameters in simulations. In this paper, we describe *visitation graphs*, a novel approximation technique for the well-established visualization of steady 2D vector field ensembles using *visitation maps*. Our method allows the efficient and robust computation of arbitrary visitation maps for vector field ensembles. A pre-processing step that can be parallelized to a high degree eschews the needs to store every ensemble member and to re-calculate every time the start position of the visitation map is changed. Tradeoffs between accuracy of generated visitation maps on one side and pre-processing time and storage requirements on the other side can be made. Instead of downsampling ensemble members to a storable size, coarse visitation graphs can be stored, giving more accurate visitation maps while still reducing the amount of data. Thus accurate visitation map creation is possible for ensembles where the traditional visitation map creation is prohibitive. We describe our approach in detail and demonstrate its effectiveness and utility on examples from Computational Fluid Dynamics.

2012 ACM Subject Classification Human-centered computing → Visualization systems and tools

Keywords and phrases Uncertain flow visualization, Ensemble visualization, Visitation maps, In-situ

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.4

Funding This research was funded by the Deutsche Forschungsgemeinschaft (DFG – German Research Foundation) under contract 252408385 as part of IRTG 2057 Physical Modeling for Virtual Manufacturing.

1 Introduction

The incorporation of uncertainties into computational models is a core challenge in computational science on which quick advances have been made in the recent past. Due to growing computational ability, it has become straightforward to investigate the effects of model and parameter uncertainty through ensemble simulation. Models are realized multiple times and the ensemble of realizations is used as a basis for analysis. However, the amount of data resulting from ensemble simulations with a high number of ensemble members can be very large and of prohibitive size to be able to interactively (in terms of reaction times) visualize and analyze it. An often-used enabling methodology in this context are *in situ* techniques that derive visualization and analysis at data production time and store reduced-size artifacts such as images, videos, or other reduced representations for further inspection.

In this setting we consider the use of *visitation maps* (Figure 1). These are used to elucidate the transport behavior that is described by an ensemble of 2D vector fields. Such visitation maps are easy to use from an analyst's point of view They can be viewed as a generalization of integral curve techniques that form the basis of flow visualization. Consequently, their computation is in practice simple and, given an initial distribution, utilizes Monte Carlo sampling of trajectories across a vector field ensemble. For large (or even medium sized)

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 4; pp. 4:1–4:20

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

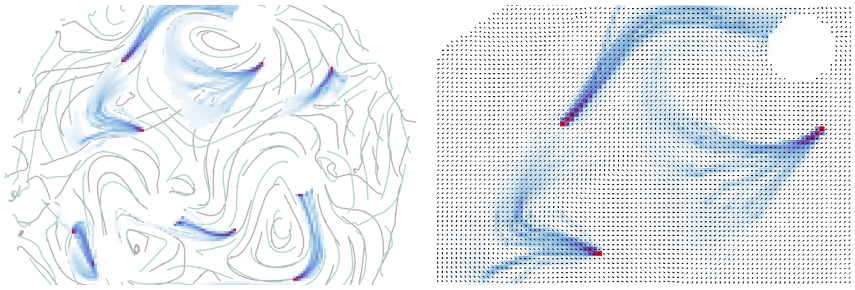


Figure 1 Left: Streamlines of two ensemble members are shown in the background to give an overall impression of the flow. Right: Glyphs calculated from the visitation graph surround the visitation maps.

vector field ensembles, however, this straightforward approach becomes prohibitively costly since running times are not adequate for interactive exploration of uncertain vector fields and all ensemble members are required for sampling.

In this work, we propose *visitation graphs* as a novel intermediate representation of the flow behavior in an ensemble of two-dimensional vector fields and show their usefulness in enabling the interactive exploration of vector field ensembles. We illustrate that visitation maps for arbitrary initial distributions can be derived quickly from the visitation graphs. Furthermore, visitation graphs give a less lossy reduction method than downsampling. Thus, visitation graphs offer the possibility to explore large ensembles interactively, with the flexibility to trade off approximation quality for representation size. In particular, our contributions are:

- After reviewing related work in Section 2, we define visitation graphs (Section 3.1) and discuss how they can be constructed from vector field ensembles (Section 3.2).
- We provide in Section 3.3 an algorithm to quickly approximate a visitation map for an arbitrary initial distribution from a visitation graph and discuss the approximation of visitation maps using shorter visitation maps.
- Details on space requirements and data reduction are given in Section 3.4 and we investigate interactive visualization modes on ensembles enabled by our approach (Section 3.5).
- The behavior of our method under variation of different parameters is examined in Section 4.
- Using a representative set of application datasets, we empirically demonstrate the usefulness of our approach and evaluate its approximation, reduction and runtime properties in Section 5.

Overall, we aim to demonstrate that visitation graphs can function ideally as artefacts of an in situ approach for vector field ensembles that still permit nearly full flexibility during post hoc exploration.

2 Background and Related Work

The present work is focused on the visualization of uncertain vector fields and vector field ensembles, as well as in situ visualization techniques. In the following, we briefly review relevant prior work and background material and discuss differences to our approach.

Visitation Maps. Visitation maps are an established and intuitive visualization to analyze distributions of particle trajectories arising from vector field ensembles or vector valued random fields.

Previous authors have defined visitation maps in an ad hoc manner as the empirical distribution within each cell in the domain obtained as the percentage of generated trajectories of a given length and with a given start point passing through the cell [4, 22, 21]. However, the resulting ad hoc calculation renders the use of visitation maps challenging due to computational effort inherent in their numerical approximation. Given an initial condition (i.e. starting location or initial distribution), the visitation map is typically estimated through direct sampling. For a faithful approximation, a high number of samples is required. For small datasets, parallel computation using for example GPUs can be leveraged to achieve interactive re-computation upon modification of the initial condition. For example, Bürger et al. demonstrated an interactive visualization of ensemble vector fields with visitation maps using GPU-based Monte-Carlo particle tracing [4]. However, for larger datasets, trajectory computation is a difficult problem and has to be handled through non-interactive out-of-core techniques [29] or parallel algorithms [34]. In contrast, our approach performs the sampling calculations in a pre-processing step that can be executed in situ, and stores the result as a compact visitation graph from which visitation maps can be derived quickly in a later state. Thus, we eschew the problem of ensembles being too large to handle interactively. In addition, a tradeoff between pre-processing time, storage requirements, and visitation map accuracy can be made.

A recent variation of visitation maps was proposed by Ferstl et al. as streamline variability plots [12]. In contrast to visitation maps, variability plots are generated by projecting confidence ellipses for obtained streamline clusters in PCA space to domain space. This yields an envelope for most obtained streamlines, together with a calculated mean streamline. For large datasets, the challenge to integrate large numbers of streamlines remains the same.

Representing Vector Fields as Graphs and Webs. Representing information on a given vector field in a graph structure, as it is done in this work, was earlier considered in the forms of Flow Graphs by Nouanesengsy et al.[29] and as Flow Webs by Xu et al. [42]. The Flow Graph contains a node for each block in an underlying grid, and each two neighboring blocks are connected via a weighted edge. The weight of the edge connecting two blocks is determined as the probability that a seeded particle in one of the blocks is transported to the other one by the flow. Afterwards, this flow graph can be used to estimate the workload for each block in parallel streamline calculation.

While Flow Graphs are an efficient approach to estimate workload, they are not suitable for visitation map approximation, as visitation maps calculated from such graphs suffer from the lack of variety in possible directions in the graph. In a Flow Graph each cell is connected only to its four direct neighbors, which results in an extremely coarse approximation of the visitation map.

In Flow Webs, regular axis-aligned sub-regions of the given domain represent nodes in the graph. By sampling streamlines backwards through the regions, links and weights between different sub-regions are determined. While Xu et al. do not consider uncertainty, our approach can be viewed as an adaptation of the Flow Web concept to ensembles to create visitation maps, based on a generalization of the Flow Connection Matrix (adjacency matrix of the Flow Web).

As opposed to the Flow Graphs by Nouanesengsy et al., Flow Graphs by Ma et al. [27] are a tool for streamline and pathline exploration of 3D flow fields. Here, nodes in the graph do not only represent spatial regions but also streamlines in the field. Edges between nodes are assigned various interpretations depending on the kind of nodes they interconnect.

Adaptive transition graphs for vector fields have been employed by Szymczak to calculate Morse connection graphs for piece wise constant vector fields on surfaces [38]. His method of refining transition graphs in an adaptive manner according to strongly connected components can be interpreted as perturbing the vector field near trivial Morse sets to remove recurrent features in the resulting Morse connection graph.

Multiple approaches employing graphs to track and visualize states and state transitions in time-varying vector fields have been proposed, for example by Gu and Wang [14] and by Jänicke and Scheuermann [19]. A recent survey on the utilization of graphs in visualization was given by Wang et al. in [39].

None of the aforementioned graph techniques are aimed at creating visitation maps and providing sufficient information on an underlying ensemble or vector valued random field to obtain adequate visitation map approximations.

Visualization of Uncertain Vector Fields. Uncertainty in vector fields has been recognized as a pressing research problem in scientific visualization for several years and still is a substantial challenge. Recent surveys can be found in [7] and [15]. Heine et al. discuss topology based visualization methods that address uncertainty in [16].

Often, uncertainties arise from multiple results of experiments with varying input parameters collected in ensembles. Examples for uncertainty visualization based on ensembles are: showing ensemble members vanishing over time [6], enabling the user to compare single members to the whole ensemble using glyphs [35] and summarizing ensemble members while highlighting outliers and median in Contour/Curve Box Plots [40, 28]. The topology of ensembles in two and three dimensions was determined by Otto et al. in [31] and [30]. Hummel et al. gave a comparative visual analysis for ensembles of time-varying vector fields using a Lagrangian framework [18]. A two dimensional comparative visual analysis was presented by Jarema et al. in [20].

Uncertainty arising from interpolation and prediction of missing measurements was treated using tubes of varying size [3], glyphs and parallel coordinates for MR spectroscopy data [11, 10], flow radar glyphs for time dependent vector fields with uncertainty given as an interval [17] and using colormapping and line glyphs for uncertain isosurfaces in geosciences [43]. Uncertain predicted multivariate data was visualized by Berger et al. using parallel coordinates and scatter plots [2]. Random fields were treated using fuzzy set theory, volume rendering on trapezoidal possibility distribution [13] and by visualizing isosurfaces in uncertain scalar fields [32, 33]. A FTLE like method was presented by Schneider et al. [36].

Random fields are a stochastic uncertainty model. Following the approach of in situ data reduction by summarizing statistics of certain properties, random fields frequently arise in in situ pre-processing. While in mathematics, the generalization of stochastic processes to higher dimensions is called *random field*, different names have been used in the visualization community up to now. Otto et al speak of *uncertain vector fields* [31], Ferstl et al. use the term *ensemble of vector fields* [12], Sevilla-Lara et. al speak of *distribution fields* in computer vision [37] and Love et al. use the more general term *spatial multivalue data* [26]. In [8], Dutta et al speak of statistical information.

Our method is suited to the uncertainty visualization of both, vector field ensembles and random fields.

Approximation of integral curves. Different approaches for the approximation of a longer particle trajectory by a sequence of (certain) flow maps have been examined. Agranovsky et al. give a two phase approach extracting a basis of known pathlines in situ and calculating arbitrary integral curves post hoc from the extracted results [1]. Two similar phases are

presented in this paper, carrying the idea to approximate longer curves using smaller ones over to visitation maps. To our knowledge, the generalization to uncertainty and thus to visitation maps in this paper is a new contribution.

In Situ Analysis and Visualization. In situ analysis, i.e. data analysis taking place at creation time while the data is still in memory, is an attractive possibility to handle huge datasets that are too large to store. In addition, slow data output is avoided and data can be pre-processed at creation time. As computational power increases and thus simulations of massive ensembles become common, the need for effective in situ processing is ever more pressing.

In this context, in situ capabilities for leading visualization libraries and tools such as Paraview [9] and VisIt [25, 41] were developed, allowing in-situ analysis and visualization of simulations.

While in situ visualization provides crucial insights in the simulation behavior at runtime, interactivity and thus exploration of the data is rarely possible. Pre-processing simulation data in situ and benefiting from reduced data in flexible and scalable post hoc analysis is a possibility to combine the advantages of in situ and post-processing. This new paradigm is illustrated for example by Childs in [5]. Several approaches were already presented that follow this paradigm. For example Lakshminarasimhan et al. proposed ISABELA for in situ sorting and error bounded compression in [23].

A further approach to reduce data in situ is to summarize statistics of properties of interest during the simulation. Afterwards suitable visualization approaches are used for data exploration. This was done recently by Dutta et al. in [8]. A recent survey was given by Li et al. in [24] grouping current research on in situ data reduction in a spectrum between lossless and very lossy techniques.

Following the in situ, post-processing combination in [5], our approach processes steady 2D vector field ensembles in situ to construct a *visitation graph*. With an upper bound for storage requirements that does not depend on the number of ensemble members, this visitation graph is an ideal structure to summarize ensembles. Furthermore, the visitation graph resolution can be chosen freely and does not have to mirror the resolution of ensemble members. This provides a means for data reduction, tailored to the task of creating visitation maps for interactive exploration of ensembles potentially too large to store (cf. Section 3.4 for additional discussion). Based on the visitation graph an interactive post hoc exploration of the ensemble using visitation maps is possible once the simulation and pre-processing are complete.

3 Method

In the following, we introduce the *visitation graph* as the core concept of our work and then proceed to illustrate its use for visitation map computation and visualization.

3.1 Visitation Graphs

We consider two-dimensional vector field ensembles or random vector fields. These are partitioned into cells of their domain of definition $\mathcal{C} = \{c_1, \dots, c_M\}$. A visitation map $\mathcal{V}$ represents for each c_j the distribution of streamline samples of integration length $T > 0$ that "hit" the cell c_j, i.e.

$$\mathcal{V}(c_j) \;:=\; P(S(X,T) \cap c_j \neq \emptyset), \tag{1}$$

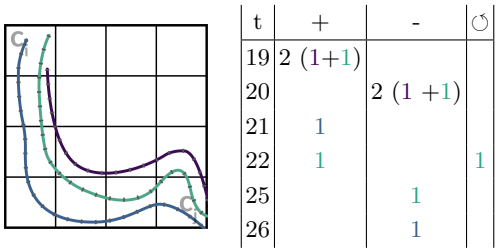


Figure 2 Visitation graph creation: Streamlines starting in cell c_i that pass by c_j are considered and all events are tracked with time point. Table entries are colored according to the triggering streamline.

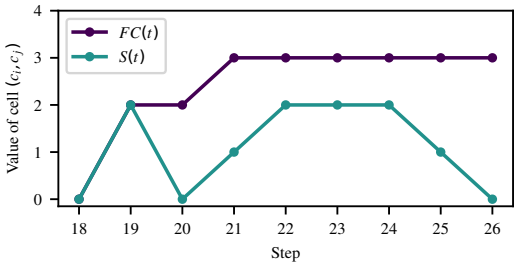


Figure 3 Values of the Flow Connection Matrix and snapshot in the example in Figure 2 (omitting normalization). $FC(t)$: number of streamlines that passed by, $S(t)$: number of streamlines that are currently in the cell.

where $S(X,T)$ is the set of all points of the streamline sample, and its seed point X is a random variable with a fixed initial distribution (in traditional visitation maps X is a uniform distribution in a single cell). In practice, this probability is computed through Monte Carlo sampling as an average over a set of streamline samples.

To speed up visitation map generation, we propose an approximation based on in situ calculated *visitation graphs*.

We define the visitation graph as a directed graph $G(V,E)$ whose nodes V are the cells of the partition, i.e. $V = \mathcal{C}$, and E contains an edge between c_i and c_j if and only if a streamline sample of length T' starting in c_i hits c_j during integration. For each edge (c_i, c_j), streamlines starting in cell c_i are considered. Their (re-)entry and exit in cell c_j are recorded.

Streamline samples are approximated as a polyline with k points, originating in every cell c_i. In practice, we (re-)use the computational grid of the ensemble as the partition $\mathcal{C}$, and approximate streamlines using a second-order Runge-Kutta scheme with fixed step size (hence k steps of step size Δt). Both of these aspects of the technique are easily modified.

Information is collected similar to the Flow Connection Matrix [42]. However, the visitation graph keeps track of more information than just the number of passing streamlines, namely the events and timepoints.

From the visitation graph, visitation maps can be assembled. We explicitly point out that, to approximate a visitation map of length T, it is not required that $k\Delta t := T' = T$ holds; in practice, streamline samples can be much shorter than the desired visitation map length, at the cost of accuracy. This allows to trade off accuracy and speed in visitation map creation for speed in visitation graph creation and storage size of the visitation graph, since shorter streamlines yield less events that need to be stored in the graph. Furthermore, the visitation graph resolution is not necessarily the same as the resolution of ensemble members. This provides a method for data reduction tailored to the task of creating visitation maps for interactive exploration of ensembles potentially too big to store. See Section 3.4 for more details on data reduction using visitation graphs.

3.2 Efficient Computation of Visitation Graphs

For each cell in the visitation graph connected to the considered start cell via an edge, a list of steps and 3-tuples is created recording the occurrence of the events of interest. This is done for each edge (c_i, c_j) by tracking how many streamlines starting in cell c_i enter cell c_j at each step (+), leave cell c_j at each step (-) and re-visited cell c_j at each step (↻). The last entry is obtained by keeping track for each cell whether the considered streamline has already entered c_j or not. An illustration of visitation graph generation can be found in Figure 2.

During visitation graph creation, the number of randomly seeded streamlines started in each cell can be determined either based on the size of the cell (as done in [42]) or on the estimated contribution to the final visitation graph. While the first option is straight forward, the latter is achieved as follows: the number of cells n_h that have been hit by any streamline that was integrated from the considered start cell is determined and a size for streamline bunches is set as parameter s_b. Every time the integration of a bunch of streamlines is completed, it is decided whether one more will be started or not using

$$\frac{n_\mathrm{h}}{s} < \frac{1.0}{s_b}. \tag{2}$$

Where s is the total number of integrated streamlines from the considered start cell. If expression 2 evaluates to false, the next bunch of streamlines is integrated, else the visitation graph creation for the considered cell stops. This stop criterion can be interpreted as follows: it is unlikely that a streamline from the upcoming bunch of streamlines passes by a cell that was not passed before since the number of passed cells per integrated streamline is smaller than one out of a bunch of streamlines.

Handling two ensemble members is independent. Hence, this calculation can be done in situ. For every ensemble member, entries in the visitation graph are updated or added. While the visitation graph can be built completely in situ, it does not have to. In case of ensemble members occupying most of the available memory storage, it might be impossible to create the whole visitation graph simultaneously. In this case, single members or even single cells can be processed in situ and then be stored. In a post processing step all partial results are combined to the final visitation graph. Depending on the available memory every possible storing frequency in between "storing for each cell or member" and "storing once at the end" can be used providing a tradeoff between postprocessing time and memory requirements.

3.3 Efficient Approximation of visitation maps from visitation graphs

Given an initial distribution of cells or a single start cell, visitation maps can be effectively calculated from the visitation graph as follows.

Using the visitation graph consisting of M nodes, two different matrices can be generated for every timepoint $t \leq T'$: The *Flow Connection Matrix* $FC(t)$ as described in [42] and the *snapshot* $S(t)$. The Flow Connection Matrix is an $M \times M$ matrix given by the adjacency matrix of the visitation graph where the weight of edge (c_i, c_j) is given by the probability that a streamline starting in cell c_i passes by cell c_j before timepoint t. The entries are calculated as the sum of the differences of first and third entry of the stored 3-tuples until t, divided by the total number of integrated streamlines starting in c_i. While the first entry provides the information how many streamlines from c_i have entered c_j, the third entry ensures that re-entering streamlines are not counted twice. Note that Xu et al. generated the Flow Connection Matrix only for infinitely many streamline steps, thus our approach constitutes a generalization of their graph structure.

Entry (c_i, c_j) of the $M \times M$ dimensional *snapshot* matrix $S(t)$ specifies the probability that particles starting in c_i are in c_j after integration time t. It is calculated as the sum of the difference of first and second entry of the stored 3-tuples until t, divided by the total number of streamlines starting in c_j. Thus every time a streamline enters c_j, the snapshot entry increases by one and as the streamline leaves c_j, the snapshot entry decreases by one. This is done regardless of re-visiting. Examples for the calculation can be found in Figure 3.

Note that when generating Flow Connection or snapshot matrices from the visitation graph, the total number of streamlines started in the considered initial cell is required. Estimating the number of required streamlines, this number varies between cells. However, it

is not necessary to store this number separately since every streamline is recorded as entering at timepoint 0 in the start cell itself. Thus the total number of streamlines per start cell is easily available in the visitation graph.

Having an integration length T' that is independent of the final visitation map length T, the desired visitation map might have more or less steps than have been integrated while pre-processing. Being able to generate a visitation map based on streamlines shorter than the final map provides a tradeoff between accuracy on one side and shorter pre-processing time and storage savings on the other side. Hence two scenarios for visitation map creation exist: Either $T \leq T'$ or $T > T'$. If $T \leq T'$, the visitation map starting in cell c_i is given by the row representing c_i in the Flow Connection Matrix. That is the exact visitation map value $\mathcal{V}(c_j)$ is given in entry (c_i, c_j) of $FC(T)$. Considering possibly multiple start cells with a start distribution, a M-dimensional start vector v_0 is created holding the start probability for each start cell. This vector is then multiplied with $FC(T)$. Each entry of the resulting vector represents one cell in the grid and holds the visitation map values. The resulting visitation map gives the identical result as a traditional visitation map based on ad hoc created streamlines advancing for time T. If, on the other hand the desired visitation map length exceeds the number of pre-processing steps ($T > T'$), the visitation map is not given in the Flow Connection Matrix but needs to be "assembled" as follows from multiple $FC(t_i)$ where $\sum_i t_i = T$.

The *law of total probability* states that if $\cup B_i = \Omega$ is a countable partition of the entire sample space, then for any event A it holds:

$$P(A) = \sum_i (P(A \mid B_i)P(B_i) \tag{3}$$

For clarity of notation let in the following S, A, and B be arbitrary cells in $\mathcal{C}$. Considering the partition consisting of events

$A_t^S :=$ After time t, the considered streamline starting in cell S ends in cell A

for the fixed integration time in pre-processing T' the law of total probability gives

$$P(B_T^S) = \sum_{A \in \mathcal{C}} P(B_T^S \mid A_{T'}^S)P(A_{T'}^S) \tag{4}$$

Where the event B_T^S is an event considering streamlines after time $T > T'$, thus no calculations for this event have been performed during pre-processing. To estimate $P(B_T^S)$, the conditional probability $P(B_T^S \mid A_{T'}^S)$ is approximated using $P(B_{T-T'}^A)$ in equation 4. So the probability that the considered streamline that starts in S ends in B after time T while ending in A after time $T' < T$ is approximated by the probability that a considered streamline starting in A ends in B after $T - T'$ steps (See Figure 4 for further explanation). This gives

$$P(B_T^S) \approx \sum_{A \in \mathcal{C}} P(B_{T-T'}^A)P(A_{T'}^S). \tag{5}$$

Since $P(B_{T-T'}^A)$ considers an integration time strictly smaller than T, the considered number of steps is potentially smaller than the number of pre-processing steps until time T' such that pre-processed results are available. If however $T - T' > T'$, the same approximation is repeated until the integration time is smaller or equal to T'.

This approximation in general overestimates $P(B_T^S \mid A_{T'}^S)$. Equality is only given if $P(A_{T'}^S) = 1$ since then $P(B_{T-T'}^A)$ is independent of the point in cell A reached by the streamline from S in time T'. Thus every approximation step will induce an error in the

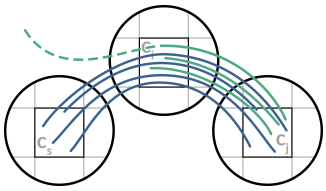


Figure 4 The probability of a streamline path $c_s \to c_i \to c_j$ in time T while in c_i after T' is approximated by the probability of the streamline path $c_i \to c_j$ in $T - T'$. This leads in general to overestimation since every streamline path $c_s \to c_i \to c_j$ is also a path $c_i \to c_j$, but not all streamlines from c_i to c_j pass by (or start in) c_s before entering c_i (f. ex the dashed streamline).

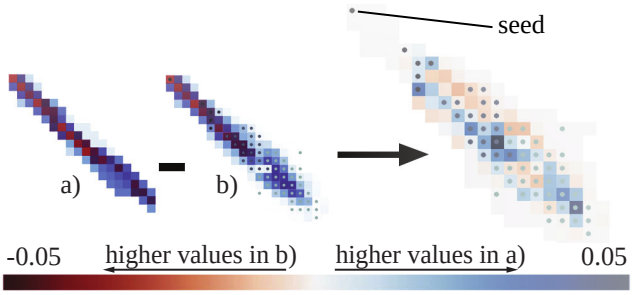


Figure 5 Traditional visitation map a), approximated based on visitation graphs b) and difference between both. $T' = 50$, $T = 160$. Nonzero entries of the start vectors v_i are marked from black to light grey for increasing i. After the first 50 (identical) steps, the approximated visitation map is re-started from all cells containing particles at step 50 (marked in grey). Resulting in an overestimation at the edges and an underestimation in the center. The colormap was chosen to present positive and negative differences and is not intended for visitation map presentation.

resulting visitation map (cf. Figure 5). This theoretical result is implemented as follows: The exact visitation map after time T' is calculated as described above using the start vector v_0. Having reached the maximal calculated step, a new start vector $v_1 = v_0 \cdot S(T')$ is calculated. Using this new start vector holding the positions and probabilities for particles starting in the considered start cell(s) under the considered start distribution after T' steps, the proceeding visitation map is calculated using $FC(T - T')$. The results are combined ensuring to not have doubled results in nonzero entries of v_1. This is repeated m times until $T - mT' \leq T'$.

Illustratively, the visitation map is followed as far as it was pre-computed, then new visitation maps are started at the end incrementally until the desired number of steps is reached.

While a smaller number of pre-processing steps induces errors in calculated visitation maps, pre-processing time becomes shorter and storage requirements of the resulting visitation graph become smaller. This tradeoff is illustrated by experiments in Section 4. Visitation map creation on the other hand becomes more expensive the more snapshots and Flow Connection Matrices are calculated, thus the calculation time for visitation maps from the visitation graph decreases with increasing pre-processing integration time T'. In general, calculation times can be reduced using parallelization: Integrating streamlines from every cell in the visitation graph to determine edges can be heavily parallelized since considering different cells is completely independent. Provided ensemble members reside on different nodes of a cluster, these members can be processed independently on their nodes. The final visitation graph can then be assembled using the parallel reduction approach resulting in logarithmic time savings. In an informal experiment, we observed nearly ideal speedup (7.96x

for 8 CPUs) for streamline integration. The assembly of 10 partial results was executed on 2 CPUs with a speedup of 1.2. In addition, the calculation of Flow Connection and snapshot matrices for visitation map creation can be naturally parallelized for every entry in the resulting matrix. While these matrices are potentially of very high rank, the fact that entries of $FC(t)$ and $S(t)$ can be calculated independently can be used to generate only rows that are required for the computation (that is the rows whose entries in the start vector are nonzero) and save them in a sparse format. Especially for the first steps in visitation map creation where the start vector contains only the chosen start cells, the number of needed rows is very small compared to the total number of rows. If however the resulting matrices become too large, a shorter visitation map or a coarser visitation graph need to be considered.

3.4 Space Requirements and Data Reduction

Visitation graphs are an optimal way to store, and potentially reduce, data in order to explore the data with visitation maps later on. The maximal storage requirements of the visitation graph representation are determined by the ensemble resolution $m \times n$ and T'. As such, they are independent of the number of ensemble members. An upper bound for the number of outgoing edges is $m \cdot n$, which is far from being tight. In reality, the number of outgoing edges is much smaller since it is impossible that every cell in the grid is passed by streamlines from every other cell. Also edges are only possible between cells that can be reached from another by a streamline of set length, thus only cells in a given, field dependent radius around each cell can be connected. At most $T' \times 4$ numbers need to be stored per edge. This upper bound will not be reached in realistic examples and most of the information stored for edges will contain far less than T' tuples (see Section 5). Thus, facing ensembles with a high number of members compared to the number of cells per ensemble, the visitation graph will save space without any additional data reduction. For other ensembles, multiple options leading to data reduction are available:

Visitation graph resolution. The visitation graph resolution is independent of the original resolution of the ensemble. Thus, a coarser resolution can be chosen. While a coarse visitation graph contains only nodes corresponding to a coarser grid, it is based on streamlines that are created based on the high resoluted ensemble. Carrying over the information from the fine grid to the coarse one, coarse visitation graphs are superior to simple downsampling of the grid. See experimental results in Section 4 for a comparison.

Timestep selection. Depending on the application, it is often not necessary to have access to visitation maps for every possible timestep. To exploit this, the temporal resolution of available visitation maps can be restricted. Defining a frequency F at which visitation maps should be available, events don't need to be stored for every timestep but for every frequency cycle. All events occurring between two cycles are summed up in one entry, reducing storage requirements since only every $F - th$ entry is stored. This reduction approach does not affect accuracy of visitation maps at available time steps.

Streamline length. Shorter streamline length at creation time of the visitation graph results in a lower number of events and thus reduces the storage requirements of the visitation graph. While it is still possible to generate longer visitation maps from the resulting visitation graph, the accuracy is reduced after every integration time of T'.

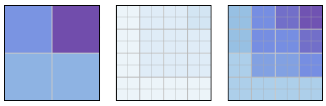


Figure 6 Appearance of the same area considering different grid resolutions. Left: coarse resolution, middle: no color compensation, right: color compensation. In every case the sum of the probabilities over all cells is one and the distribution is equal.

Experiments on data reduction using visitation graphs as well as experiments on storage requirements and their dependency on the number of ensemble members can be found in Section 4. An application of data reduction can be found in Section 5.

3.5 Application to Visualization

As outcome of the previous results, visitation maps from arbitrary start points with arbitrary initial distributions can be generated effectively from visitation graphs. This heavily contrasts with traditional visitation maps generated in ad hoc manner from integrated streamlines. They trigger a complete re-computation every time the start point is changed and considering multiple start cells with a given distribution results in vast computational effort. In addition, traditional streamline computation is always based on the whole ensemble, making this visualization approach unfeasible for ensembles with too many members to store. In our approach, to interactively create visitation maps from arbitrary start points, only the visitation graph needs to be stored. There is no need to store single ensemble members. Providing a tradeoff between storage requirement and accuracy, the visitation graph renders the exploration of ensembles possible that are prohibitive to store due to their size. Provided a visitation graph, two different modes for visitation map creation are available:

Standard mode: A single start point is selected. The visitation map of the desired length is calculated or approximated depending on T'.

Brush mode: Multiple start points following an initial distribution are given. The visitation map is calculated for all start points at once. Different kernel functions are available as possible start distributions, they are scaled such that the integral over all initial cells is one.

While for regular grids the visualization of visitation maps (ad hoc or visitation graph based) is a straight forward mapping of scalars to colors, additional challenges arise when considering grids containing differing cell sizes or the same grid is evaluated in different resolutions.

The probability that a streamline passes by a grid cell depends on the cell's size. Since the sum of the probabilities over all cells in an area remains constant independent of the resolution, small cells obtain smaller probability values in the visitation map than bigger cells. While from the mathematical point of view, this result is correct and intuitive, the resulting visualization might be misleading. Areas with smaller cells might appear less likely to be visited than coarser areas with the same probability. To handle this effect when plotting the visitation map, a compensation for the cell size can be used. The probability value assigned to each cell is multiplied by the size of the largest cell in the grid divided by the size of considered cell. This yields an intuitive visualization, see Figure 6. However the plotted values no longer represent the actual probabilities, so both visualization options are accessible in our implementation.

To give an orientation in the explored vector field ensemble, visitation maps can be combined with any other suitable static visualization as background. In the case of not having stored single ensemble members, this is hardly possible with common techniques.

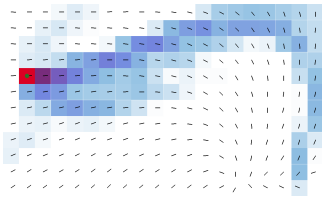


Figure 7 Glyphs indicating the average direction of connected cells in the visitation graph.

To be still able to give an impression of the underlying field, we developed visualization techniques based on the visitation graph, intended to support the visitation map visualization:

Glyph visualization: glyphs pointing towards the average direction of connected cells in the Flow Connection Matrix weighted with their probability can be drawn. That is, for every nonzero entry (c_i, c_j) in the Flow Connection Matrix, the midpoint of cell c_j contributes with weight $FG(t)(c_i, c_j)$ to the direction the glyph in cell c_i will point to. See Figure 7 for an example.

Degree visualization: In- and out- degree of each cell in the visitation graph is plotted using different colors. Like this, areas with large variance and areas that are passed by many different streamlines are visible, giving an orientation which areas might be interesting in to explore. See Figure 14 for an example.

While exploration of vector field ensembles is the main purpose of our approach, it can be employed to visualize random fields as well.

Given a probability space $(\Omega, \mathcal{F}, P)$, a random field of dimension $d \in \mathbb{N}$ is a family of random variables $\{Y(x, \cdot)\}_{x \in \mathbb{R}^d}$. A trajectory in a random field can be defined analogously to the certain case by an ordinary differential equation with a random field on the right hand side, which makes the equation a random ordinary differential equation (RODE).

$$\frac{dX_i(t,\omega)}{dt} = Y_i(X(t,\omega), t, \omega) \qquad \text{with } t \geq t_0, \omega \in \Omega, i = 1 \dots d. \tag{6}$$

RODEs can be solved using standard numerical approaches such as Runge-Kutta methods or multi-step methods. Thus, instead of integrating streamlines in every ensemble member, multiple streamlines are integrated using samples of the underlying random field.

4 Experiments

To evaluate the behavior of the system, different tests have been conducted on a standard workstation PC. The implementation was done in Python. Unless otherwise noted, a 50×50 testing ensemble containing 50 members created from normal distributions with constant $\mu = (1, 1)$ and varying correlation between -0.5 and 0.5 was considered for all tests in this section. The cell size was uniform and T' was chosen as 60 while $T = 100$.

Figure 8 illustrates the tradeoffs resulting in the generation of the visitation graph: a smaller number of streamline steps in pre-processing results in lower accuracy of the resulting visitation maps while time for visitation graph creation and storage requirements drop. Conversely, the longer the pre-processing integration time, the more accurate is the result while pre-processing runtime and storage requirements increase. The runtime for visitation map generation depends on T': With fewer timesteps stored in the graph, more calculations have to be executed to obtain the visitation map approximation (cf. Section 3.3). Thus, time needed for visitation map creation decreases with increasing accuracy. All runtimes are measured on a purely serial execution of the code. As ground truth, the visitation

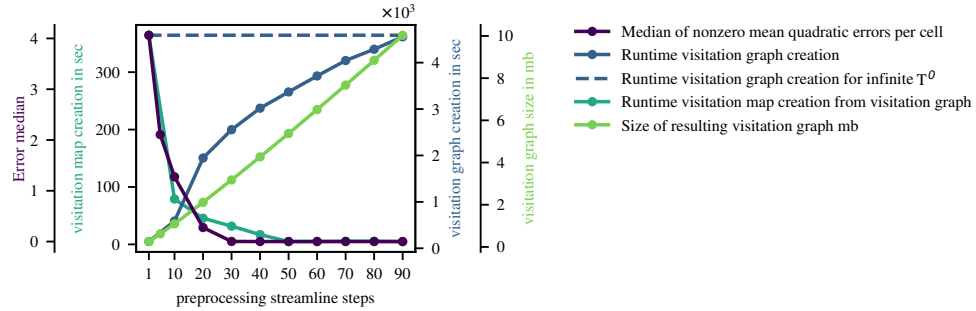


Figure 8 Visitation graph tradeoffs: With increasing T', accuracy, storage requirements and pre-processing time increase while the required time to create visitation maps decreases. The median of nonzero errors is shown since low errors are far more common than high errors.

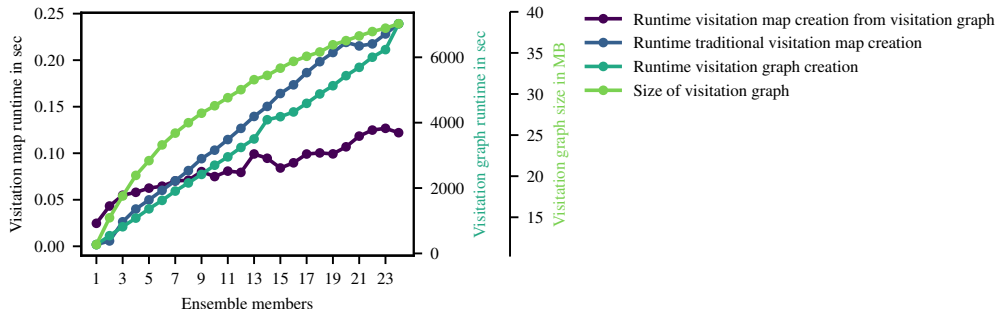


Figure 9 The more members are considered, the more beneficial is our method. The difference in runtime increases and already for very few ensemble members it outperforms the traditional ad-hoc creation. Storage requirements for visitation graphs converge towards a fix maximal size. Visitation graph creation grows linearly with the number of processed ensembles.

graph with infinite T' and corresponding visitation maps are considered. That is every generated streamline is pursued until it either ends in a critical point or leaves the grid. For less streamline steps, the ground truth visitation graph is pruned at the desired step to avoid differences resulting from randomized streamline seeding. The mean squared error is calculated per cell as the difference between traditional visitation map starting in this cell and the visitation map generated from the visitation graph. As the number of pre-processing steps reaches the number of steps taken in the visitation map, the error drops to zero. No assembly is required and the exact visitation map is calculated.

In addition to the advantage of in situ pre-processing obviating the need to store every ensemble member to be able to calculate visitation maps, Figure 9 illustrates that already for a small number of ensemble members, the visitation map creation from the visitation graph is faster than the ad hoc creation. The larger the ensemble, the more beneficial is our method. This observation is strengthened when considering multiple start cells (cf. Figure 10). To obtain accurate results for traditional visitation maps with multiple start cells under a delocalized start distribution, even more realizations are necessary to achieve adequate sampling.

To examine visitation graphs as an alternative to downsampling, we executed a further experiment: Our approach using coarse visitation graphs to obtain visitation maps was compared with the traditional approach to first sample down the data, then store it and create visitation maps in the traditional way. All approaches result in visitation maps of the same output resolution. A real-world and an analytical example were tested. The resolution was reduced from 168×168 to 80×80 and 336×168 to 160×80 respectively. We compared:

Our Method: A visitation graph of output resolution is calculated on high resoluted input data. Using the visitation graph visitation maps are created.

Traditional: High resoluted input data is downsampled to output resolution. Visitation maps are calculated in the traditional way.

Ground Truth (Test on analytical data): Exact streamlines in the analytical field are used to create exact visitation maps of the given output resolution.

Ground Truth (Test on real-world data): Visitation maps are calculated in the traditional way on high resoluted input data, the final visitation map is sampled in the given output resolution.

Figure 11 holds the results for the real-world example. Because of the exact ground truth, errors of our method in the analytical example were higher. While they reside in the same range as the errors of the traditional approach, the maximal deviation only exceeds the one of the traditional approach after $T = 160$. For $T = 100$ again the error is much smaller than the one generated by traditional approximation. For $T \in [110, 160]$ our method gives a similar mean deviation and a smaller maximal deviation. So again our method outperforms the traditional approach not even for $T = T'$ but also for additionally approximated visitation maps for $T > T'$ up to a limit.

5 Results and Applications

In this section, results for real-world applications are given. While our contribution is a new calculation method for visitation maps, using visitation maps for visualization is well-established. Hence, this section aims more for giving details on applications of our method than on illustrating usefulness of visitation maps in general. The following datasets were used:

Industrial Stirring: Mixing in a stirring apparatus was simulated for 20 slightly differing viscosities resulting in a 2D flow field ensemble of size 168×168. The device consists of two counter-rotating pairs of mixing rods that stir a medium in a cylindrical tank.

Convection: Flow around a hot pole was simulated for 30 slightly perturbed initial velocity conditions at the bottom of the domain. The resulting 2D flow field ensemble is of size 128×256. Material at rest is heated around the pole, begins to rise, and forms a plume.

Cavity Flow: Laminar, in-compressible flow in a two-dimensional square domain where one border is moving with $1\frac{m}{s}$ was simulated. The Reynolds number of the simulated liquid is increased by one between each member generation, resulting in 1990 ensemble members with Reynolds numbers between 10 and 2000. The resolution is 1000×1000.

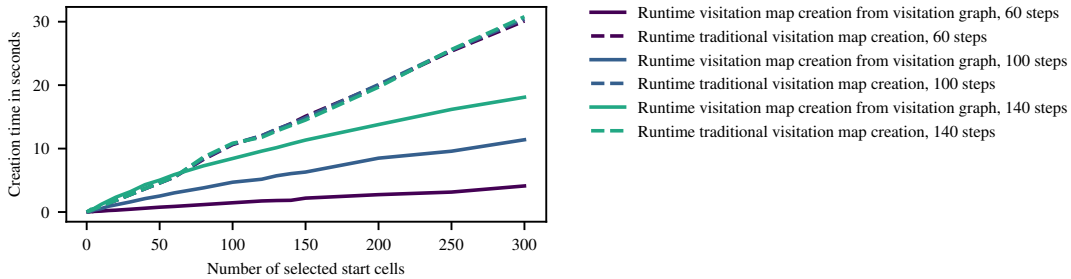


Figure 10 The creation time of visitation maps starting in multiple start cells. Visitation maps are faster generated from visitation graphs. The difference becomes larger with increasing number of start cells. In pre-processing, streamlines of 60 steps were calculated. Thus to calculate visitation maps with 100 and 140 steps, two resp. three calculation steps are needed. Intervals requiring the same number of steps result in similar generation times.

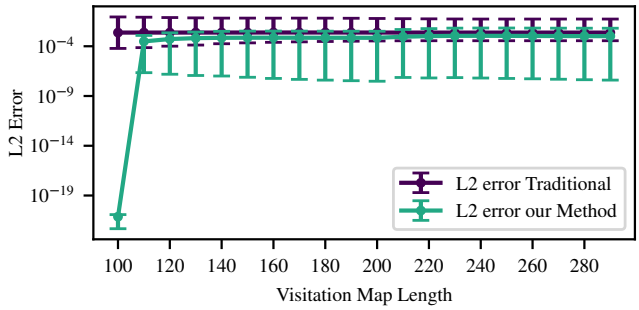


Figure 11 Visitation graphs are an optimal way to reduce data resolution and generate visitation maps afterwards: $T' = 100$, thus for a visitation map length of 100, our method equals the ground truth. Also for higher $T > T'$, our method outperforms the traditional approach. The increase in the error of our method at a visitation map length of 220 is to be expected every time T is increase d by T'. The error of the traditional method is higher and varies more.

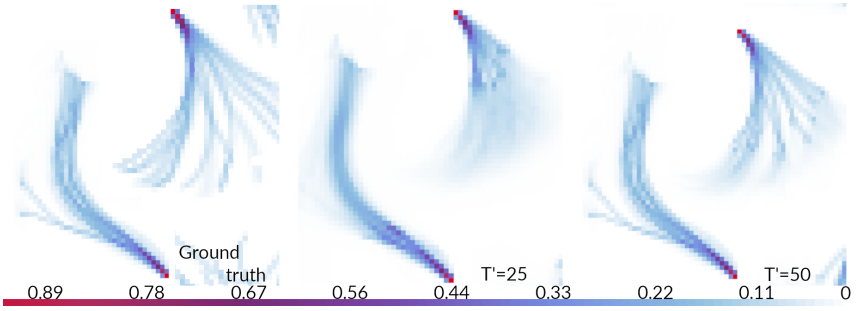


Figure 12 Results for $T' = 25$ (center) and $T' = 50$ (right) are compared with the ground truth (left). The approximation start after 25 and 50 steps respectively is clearly visible.

In all ensembles a fixed timepoint was investigated using visitation maps based on a visitation graph.

Industrial Stirring simulation. This example illustrates the approximation of visitation maps using visitation graphs based on shorter streamlines and different choices for T'.
Two pre-processing integration times were used. Serial runtime for $T' = 50$ with about 40 minutes per member is significantly higher than runtime for $T' = 25$ (about 20 minutes). The number of realizations was estimated for the latter execution. Based on a bunch size of 10, cells with very small vectors, especially the ones where no data exists, were start point for much less streamlines than in the non-automated calculation with 100 realizations for every cell. An exploratory visualization of the stirring simulation is given in Figure 1. Combining multiple start points with glyph visualization or streamline visualization of multiple members gives an overall impression. Additional initial points can be selected interactively. Like this, interesting regions, for example the flow behavior around the mixing rods can be examined easily by starting visitation maps close to them. With a pre-processing streamline length of 50, visitation maps of length 100 were generated. In Figure 12, visitation maps for $T' = 25$ and $T' = 50$ are compared. The error induced after the first approximation step is clear to see. Still, the tradeoff between pre-processing time and accuracy is profitable for exploration. The early approximated visitation maps are as useful as the ones based on $T' = 50$. In addition, the storage requirement of the visitation graph drops from 380MB to 118MB. Compared to the original ensemble size is 4.5MB, this is an increase. Yet, this example does

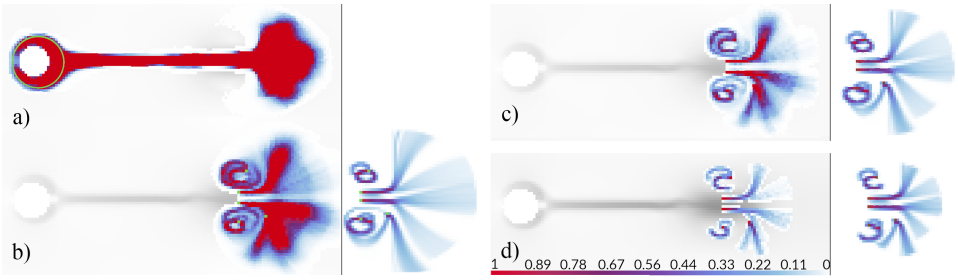


Figure 13 Convection simulation: Seeding multiple start cells around the pole (green circle) gives a first impression (a). Areas of interest are then explored using multiple single start cells (b-d). Visitation map length from b) to d) is 300,200,100. On the right, traditional visitation maps generated from 10.000 streamlines are given. For $T = T'$ differences result from the different number of streamlines. For $T > T$, the required approximation steps blur the result.

not aim on reduction of the data size. With 20 members of size 168×168, the number of ensemble members is not large compared to the number of cells. The average numbers of edges are 163.4 and 72.0 for $T' = 25$, 50 respectively, confirming that in real world examples the upper bound of $m \times n = 28,224$ is vastly overestimating the number of outgoing edges. The average number of stored 4-tuples per edge is 10.3 and 7.3 respectively which is again much smaller than the upper bound of T' in both cases.

Convection simulation. In this example we illustrate interactive exploration possibilities using visitation maps that are based on visitation graphs. A visitation graph with $T' = 100$ was generated for a single timestep of the convection simulation (Figure 13). Having detected the area of turbulent behavior using seeds around the pole, it can be explored by interactively adding multiple start points. Figures 13 a) and b) to d) show this process using different visitation map lengths; for orientation the average velocity over all ensemble members is given in the background. Processing one member took 145 minutes. Again, the upper bound of $m \times n = 32,768$ outgoing edges per cell is much higher than the real number which is 18.8 on average. 14.8 timesteps are stored per edge on average which is again much less than the upper bound $T' = 100$. This results in raw data of 72.8MB compared to original data size of 5.9MB. While our approach is not able to reduce the data in this case with a relatively low number of ensemble members, the speedup in visitation map creation is still given.

Cavity Flow. This example illustrates the data reduction abilities of visitation graphs. On the cavity flow dataset with a high number of members, a data-reducing visitation graph of resolution 100×100 was created with a timestep-storing frequency of 10 and T'=20. While creating visitation maps based on the original resolution might be impossible, the resulting visitation graph is able to retain information from the original resolution and still reduce the data. With 8 cores and a speedup factor close to 8, the calculation time of the visitation graph is about 140sec per member for $T' = 20$ and 100 streamlines per cell. Instead of the 30.2GB large ensemble, the visitation graph with 43.7MB can be stored and used to create visitation maps. On average, 131.67 outgoing edges were stored per cell and on each edge on average 1.15 timesteps (this number is reduced due to the chosen storing frequency). In Figure 14, an example for incremental exploration of the field with degree visualization background is given. Visitation maps generated from visitation graphs and in the traditional way are compared and a visitation map for the fine resolution is given.

6 Discussion and Conclusion

Vector field ensembles can be explored in an intuitive way using visitation maps. Traditional ad hoc calculation of visitation maps requires every ensemble member to be stored and sampled. Thus it can not be applied to ensembles that have a large number of members. In addition, every change of the seed point (or initial distribution) requires a full re-computation of the visitation map. And an initial distribution that is not strongly localized, results in even more time expensive computation. Visitation graphs provide solutions to these problems by isolating the sampling in in situ pre-processing while allowing the computation of visitation maps in an interactive fashion also for large ensembles and delocalized initial distributions. Costly calculation time is shifted to in situ pre-processing.

While the speed up of visitation map generation is present already for small ensembles, our method provides substantial advantages for ensembles with many members. For these ensembles, storage savings can be accomplished and visitation map generation speed up is quite significant. Multiple additional storage saving techniques are available for visitation graphs, allowing in-situ data reduction that is superior to other approaches with respect to visitation map creation.

The visitation graph stores information about all events of all generated streamline samples. Depending on spatial resolution, storage requirements can still be significant. A possible aggregation of the data in addition to a set storing frequency is part of our future work. Furthermore, parallelization (potentially on GPUs) can provide dramatically better runtimes for visitation graph generation and will be further examined in future work. Further potential for runtime reduction lies in re-using of generated streamlines during visitation graph generation.

Representing vector field ensembles as their corresponding visitation graphs offers a plethora of opportunities. Graph algorithms and clustering could be applied (e.g. in similarity to [38]), to exploit all strengths of this representation. Considering multiple time-steps of time dependent fields, results could be compared by comparing the visitation graphs, and missing time-steps could be interpolated on the graph level. Further investigations in these very promising areas are planned. Concerning visualization, different techniques could be applied to present the calculated visitation map such as isocontours or using transparency.

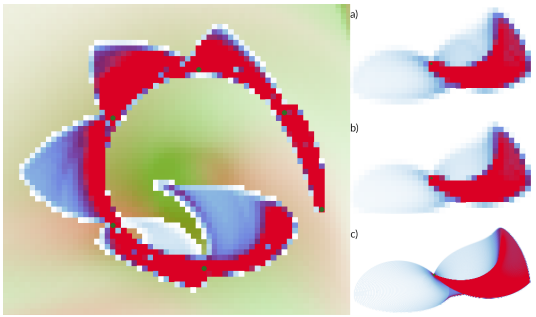


Figure 14 The cavity flow simulation. Left: Interactive exploration of the huge ensemble is possible. In the background, in- and out-degree of the cells is plotted in red and green respectively, indicating interesting areas. Seeding multiple start cells, the swirl in the data can be easily discovered and isolated. Right: Visitation maps from the same point, created with different approaches: a) our approach (0.06 sec) b) traditional based on the original ensemble (ground truth) (129,21 sec), c) traditional with original resolution (19,322.96 sec).

Finally, generalization of visitation graphs to 3D random fields will be part of our future work. While the basic technique of visitation graphs can be transferred quite naturally to 3D, the increasing number of grid cells will be the main challenge. The data reduction techniques presented in this paper seem to be promising to deal with this challenge. In addition, a link between visitation graphs and graphs used for 3D space partitioning is an exciting research topic we are going to examine. Furthermore the generalization of visitation maps to 3D requires an evaluation of existing 3D scalar field visualization techniques for this special purpose.

References

1 A. Agranovsky, D. Camp, C. Garth, E. W. Bethel, K. I. Joy, and H. Childs. Improved post hoc flow analysis via lagrangian representations. In *2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV)*, pages 67–75, November 2014. doi:10.1109/LDAV.2014.7013206.

2 W. Berger, H. Piringer, P. Filzmoser, and E. Gröller. Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction. *Computer Graphics Forum*, 30(3):911–920, 2011. doi:10.1111/j.1467-8659.2011.01940.x.

3 Ryan A. Boller, Scott A. Braun, Jadrian Miles, and David H. Laidlaw. Application of uncertainty visualization methods to meteorological trajectories. *Earth Science Informatics*, 3(1):119–126, June 2010. doi:10.1007/s12145-010-0052-5.

4 Kai Bürger, Roland Fraedrich, Dorit Merhof, and Rüdiger Westermann. Instant visitation maps for interactive visualization of uncertain particle trajectories. In *IS&T/SPIE Electronic Imaging*, pages 82940P–82940P. International Society for Optics and Photonics, 2012.

5 Hank Childs. Data exploration at the exascale. *Supercomputing frontiers and innovations*, 2(3):5–13, 2015.

6 Jonathan Cox, Donald House, and Michael Lindell. Visualizing uncertainty in predicted hurricane tracks. *International Journal for Uncertainty Quantification*, 3(2):143–156, 2013.

7 John Dill, Rae Earnshaw, David Kasik, John Vince, and Pak Chung Wong. *Expanding the frontiers of visual analytics and visualization*. Springer, 2012.

8 Soumya Dutta, Chun-Ming Chen, Gregory Heinlein, Han-Wei Shen, and Jen-Ping Chen. In situ distribution guided analysis and visualization of transonic jet engine simulations. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):811–820, 2017.

9 Nathan Fabian, Kenneth Moreland, David Thompson, Andrew C Bauer, Pat Marion, Berk Gevecik, Michel Rasquin, and Kenneth E Jansen. The paraview coprocessing library: A scalable, general purpose in situ visualization library. In *Large Data Analysis and Visualization (LDAV), 2011 IEEE Symposium on*, pages 89–96. IEEE, 2011.

10 D. Feng, L. Kwock, Y. Lee, and R. Taylor. Matching visual saliency to confidence in plots of uncertain data. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):980–989, November 2010. doi:10.1109/TVCG.2010.176.

11 David Feng, Lester Kwock, Yueh Lee, and Russel M. Taylor. Linked exploratory visualizations for uncertain mr spectroscopy data. *Proc.SPIE*, 7530:7530–7530–12, 2010. doi:10.1117/12.839818.

12 Florian Ferstl, Kai Bürger, and Rüdiger Westermann. Streamline variability plots for characterizing the uncertainty in vector field ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):767–776, 2016.

13 N. Fout and K. L. Ma. Fuzzy volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2335–2344, December 2012. doi:10.1109/TVCG.2012.227.

14 Yi Gu and Chaoli Wang. Transgraph: Hierarchical exploration of transition relationships in time-varying volumetric data. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2015–2024, 2011.

15 Charles D Hansen, Min Chen, Christopher R Johnson, Arie E Kaufman, and Hans Hagen. *Scientific visualization: uncertainty, multifield, biomedical, and scalable visualization*. Springer, 2014.

16 C. Heine, H. Leitte, M. Hlawitschka, F. Iuricich, L. De Floriani, G. Scheuermann, H. Hagen, and C. Garth. A survey of topology-based methods in visualization. *Computer Graphics Forum*, 35(3):643–667, 2016. doi:10.1111/cgf.12933.

17 Marcel Hlawatsch, Philipp Leube, Wolfgang Nowak, and Daniel Weiskopf. Flow radar glyphs - static visualization of unsteady flow with uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1949–1958, 2011.

18 M. Hummel, H. Obermaier, C. Garth, and K. I. Joy. Comparative visual analysis of lagrangian transport in cfd ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2743–2752, December 2013. doi:10.1109/TVCG.2013.141.

19 Heike Jänicke and Gerik Scheuermann. Visual analysis of flow features using information theory. *IEEE Computer Graphics and Applications*, 30(1):40–49, 2010.

20 M. Jarema, J. Kehrer, and R. Westermann. Comparative visual analysis of transport variability in flow ensembles. *Journal of WSCG.*, 24(1):25–34, 2016. URL: http://hdl.handle.net/11025/21642.

21 D. K. Jones. Tractography gone wild: Probabilistic fibre tracking using the wild bootstrap with diffusion tensor mri. *IEEE Transactions on Medical Imaging*, 27(9):1268–1274, September 2008. doi:10.1109/TMI.2008.922191.

22 Derek K. Jones and Carlo Pierpaoli. Confidence mapping in diffusion tensor magnetic resonance imaging tractography using a bootstrap approach. *Magnetic Resonance in Medicine*, 53(5):1143–1149, 2005. doi:10.1002/mrm.20466.

23 Sriram Lakshminarasimhan, Neil Shah, Stephane Ethier, Scott Klasky, Rob Latham, Rob Ross, and Nagiza F. Samatova. Compressing the incompressible with isabela: In-situ reduction of spatio-temporal data. In Emmanuel Jeannot, Raymond Namyst, and Jean Roman, editors, *Euro-Par 2011 Parallel Processing*, pages 366–379, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

24 S. Li, N. Marsaglia, C. Garth, J. Woodring, J. Clyne, and H. Childs. Data reduction techniques for simulation, visualization and data analysis. *Computer Graphics Forum*, 37(6):422–447, 2018. doi:10.1111/cgf.13336.

25 Jay F Lofstead, Scott Klasky, Karsten Schwan, Norbert Podhorszki, and Chen Jin. Flexible io and integration for scientific codes through the adaptable io system (adios). In *Proceedings of the 6th international workshop on Challenges of large applications in distributed environments*, pages 15–24. ACM, 2008.

26 A. L. Love, A. Pang, and D. L. Kao. Visualizing spatial multivalue data. *IEEE Computer Graphics and Applications*, 25(3):69–79, May 2005. doi:10.1109/MCG.2005.71.

27 Jun Ma, Chaoli Wang, and Ching-Kuang Shene. Flowgraph: A compound hierarchical graph for flow field exploration. In *2013 IEEE Pacific Visualization Symposium (PacificVis)*, pages 233–240. IEEE, 2013.

28 Mahsa Mirzargar, Ross T Whitaker, and Robert M Kirby. Curve boxplot: Generalization of boxplot for ensembles of curves. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2654–2663, 2014.

29 Boonthanome Nouanesengsy, Teng-Yok Lee, and Han-Wei Shen. Load-balanced parallel streamline generation on large scale vector fields. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1785–1794, 2011.

30 M. Otto, T. Germer, and H. Theisel. Uncertain topology of 3d vector fields. In *2011 IEEE Pacific Visualization Symposium*, pages 67–74, March 2011. doi:10.1109/PACIFICVIS.2011.5742374.

31 Mathias Otto, Tobias Germer, Hans-Christian Hege, and Holger Theisel. Uncertain 2d vector field topology. *Computer Graphics Forum*, 29(2):347–356, 2010. doi:10.1111/j.1467-8659.2009.01604.x.

32 Tobias Pfaffelmoser, Matthias Reitinger, and Rüdiger Westermann. Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields. *Computer Graphics Forum*, 30(3):951–960, 2011. doi:10.1111/j.1467-8659.2011.01944.x.

33 K. Pothkow and H. C. Hege. Positional uncertainty of isocontours: Condition analysis and probabilistic measures. *IEEE Transactions on Visualization and Computer Graphics*, 17(10):1393–1406, October 2011. doi:10.1109/TVCG.2010.247.

34 Dave Pugmire, Hank Childs, Christoph Garth, Sean Ahern, and Gunther H. Weber. Scalable computation of streamlines on very large datasets. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, SC '09, pages 16:1–16:12, New York, NY, USA, 2009. ACM. doi:10.1145/1654059.1654076.

35 Jibonananda Sanyal, Song Zhang, Jamie Dyer, Andrew Mercer, Philip Amburn, and Robert Moorhead. Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1421–1430, 2010.

36 Dominic Schneider, Jan Fuhrmann, Wieland Reich, and Gerik Scheuermann. *A Variance Based FTLE-Like Method for Unsteady Uncertain Vector Fields*, pages 255–268. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. doi:10.1007/978-3-642-23175-9_17.

37 L. Sevilla-Lara and E. Learned-Miller. Distribution fields for tracking. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1910–1917, June 2012. doi:10.1109/CVPR.2012.6247891.

38 Andrzej Szymczak. Morse connection graphs for piecewise constant vector fields on surfaces. *Computer Aided Geometric Design*, 30(6):529–541, 2013.

39 Chaoli Wang and Jun Tao. Graphs in scientific visualization: A survey. *Computer Graphics Forum*, 36(1):263–287, 2017. doi:10.1111/cgf.12800.

40 Ross T Whitaker, Mahsa Mirzargar, and Robert M Kirby. Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2713–2722, 2013.

41 Brad Whitlock, Jean M. Favre, and Jeremy S. Meredith. Parallel in situ coupling of simulation with a fully featured visualization system. In *Proceedings of the 11th Eurographics Conference on Parallel Graphics and Visualization*, EGPGV '11, pages 101–109, Aire-la-Ville, Switzerland, Switzerland, 2011. Eurographics Association. doi:10.2312/EGPGV/EGPGV11/101-109.

42 Lijie Xu and Han-Wei Shen. Flow web: A graph based user interface for 3d flow field exploration. In *IS&T/SPIE Electronic Imaging*, pages 75300F–75300F. International Society for Optics and Photonics, 2010.

43 Björn Zehner, Norihiro Watanabe, and Olaf Kolditz. Visualization of gridded scalar data with uncertainty in geosciences. *Computers & Geosciences*, 36(10):1268–1275, 2010. doi:10.1016/j.cageo.2010.02.010.

Optimized Routine of Machining Distortion Characterization Based on Gaussian Surface Curvature

Destiny R. Garcia
Mechanical and Aerospace Engineering Department, University of California Davis, CA, USA

Barbara S. Linke[1]
Mechanical and Aerospace Engineering Department, University of California Davis, CA, USA

Rida T. Farouki
Mechanical and Aerospace Engineering Department, University of California Davis, CA, USA

Abstract

Machining distortion presents a significant problem in products with high residual stresses from materials processing and re-equilibration after machining removes a large part of the material volume and is common in the aerospace industries. While many papers research on mechanisms of machining distortion, few papers report on the measurement, processing and characterization of distortion data. Oftentimes only line plot data is used to give a maximum distortion value. This paper proposes a method of measurement tool selection, measurement parameter selection, data processing through filtering and leveling, and use of Bézier Surfaces and Gaussian Curvature for distortion characterization. The method is demonstrated with three sample pieces of different pocket geometry from quenched aluminum. It is apparent that samples with machining distortion can have complex surface shapes, where Bézier Surfaces and Gaussian Curvature provide more information than the commonly used 2D line plot data.

2012 ACM Subject Classification Applied computing → Engineering

Keywords and phrases Machining distortion, Metrology, Gaussian curvature

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.5

Funding *Destiny R. Garcia*: partially funded by NSF award No. 1663341.
Barbara S. Linke: partially funded by NSF award No. 1663341.

Acknowledgements Thank you to Michael Hill, UC Davis for guiding the machining distortion research. Many thanks to Christopher D'Elia and Renan Ribeiro, both UC Davis, for designing and machining the samples. We express our sincere thanks to Jan Aurich, Benjamin Kirsch, and Daniel Weber at TU Kaiserslautern for discussing the distortion application. All above mentioned researchers are collaborators on the *NSF/DFG Collaboration to Understand the Prime Factors Driving Distortion in Milled Aluminum Workpieces*, funded through NSF Award No. 1663341 and DFG Project No. 351381681. We also thank Jörg Seewig from TU Kaiserslautern for the helpful discussions on metrology. We furthermore interact in the IRTG 2057 *Physical Modeling for Virtual Manufacturing Systems and Processes*, funded by DFG.

Nomenclature

x, y, z	Cartesian coordinates
a, b, c	constants
z_{min}, z_{max}	minimum & maximum z heights
R	range of z heights
$b_k^n(t)$	degree n Bernstein basis functions

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 5; pp. 5:1–5:17

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

$x_{\min}, x_{\max}$ minimum & maximum x coordinate
$y_{\min}, y_{\max}$ minimum & maximum y coordinate
E least–squares–fit error function
c_{kl} coefficients of least–squares function
μ, ν matrix element indices
$M_{\mu,\nu}$ least–squares matrix elements
r_μ least–squares vector elements
$\mathbf{r}(u, v)$ vector parametric surface
$\mathbf{n}$ surface unit normal vector
E, F, G surface first fundamental form coefficients
L, M, N surface second fundamental form coefficients
K surface Gaussian curvature
$f(x, y)$ functional surface
z_{rms} root–mean–square surface height

1 Introduction

In the aerospace and manufacturing industries, billions of dollars are attributed to costs from reworking, remanufacturing, and/or rejecting components that are defective due to machining distortions [3] [28]. Machining distortion is the deviation of part shape from the original intent after machining and being released from a fixture [6]. Residual stresses locked into the workpiece are a primary factor contributing to machining distortions, coming from prior material processing steps such as rolling, forging, heat treating, etc. Residual stress develops from three main mechanisms: non-uniform plastic deformation, surface modification, and material phase and/or density changes [4] [29]. After processing, the residual stresses can cause deformation due to the re-equilibration of the tensile and compressive stresses to arrive at mechanical equilibrium within the whole volume of material [4] [16]. Every year, the aerospace industry experiences significant loss in profits from part distortion [26]. Example components and processes are wing panels and other aerospace components from quenched aluminum, where deep pockets are removed by milling.

To tackle machining distortion, empirical trials and computational approaches are employed. For example, analytical models have been developed to predict the distortion of monolithic aerospace components [21]. A bending moment model for predicting shape deviation has shown useful in simple geometries [5]. A physics-based materials processing simulation revealed similar results compared to experimental data [2]. Similarly, a physics-based machining model incorporating dynamic cutting forces and tool compliance properties has been developed to predict in-process deflections along computer numeric control (CNC) machining tool paths [18]. An enhanced analytic elasto-plastic model which uses superposition of thermal and mechanical stresses, followed by relaxation procedures has been developed to predict residual stresses in machining [20]. However, in the distortion literature, basic information is missing about how distortion is practically characterized and therefore the results of experiments or modeling approaches are not easy to transfer between works of different researchers. More information is needed on the measurements themselves as well as data processing to enable easy transfer of results between distortion papers. In manufacturing research, experimentation and measurements are closely related, but they also are clearly distinct – to which measurements deserve to be viewed independently from the study of experimentation [24]. Measurements are not exact, but rather depend on multiple factors including the measurement procedure, the operator skill, the environment,

and more [1]. Measurements have an associated measurement uncertainty that is used in a central role to assess quality and quality standards [10]. There are many types of measuring principles related to production metrology including but not limited to stylus instrument, white-light interferometer, confocal microscope, focus variation microscopy, angular resolved scattering light sensor, pneumatic distance measurement, etc. [19]. There are advantages and disadvantages for each measuring principle and ultimately the measurement principle is selected based on requirements of the production.

The goal of this study is to *find a transparent way to collect, quantify, and analyze distortion data* to enhance the transferability of experimental and analytical results. The focus is on areal measurement data. An approach to characterizing machining distortion by the Gaussian curvature of a parametric Bézier surface, created by a least–squares fit, is demonstrated as a novel means to metrologically characterize distortion.

2 Metrology of distortion

Metrology in general is focused on design, maintenance, and improvements to technology to create accurate measurements [24]. Metrology has the potential to improve the capability of manufacturing processes in a production environment [25]. A measurement is obtained from a measurement process, which requires measurement variables, hardware, software, and human input to carry out the measurement [17].The introduction of measuring systems in a company or industry is often driven by the need to ensure a specific level of quality in manufacturing, using metrology for inspection purposes [25]. A measurement systems analysis or MSA, is used to characterize the measurement process [17].

2.1 Coordinate measuring machines

For many researchers the measurement instrument of interest for machining distortion research is the coordinate measuring machine (CMM). A CMM is a measuring system with a probing system and the capability to determine spatial coordinates on the surface of the measured part [8] [11] [22]. It combines measured points to form a feature using coordinate metrology. Coordinate metrology provides a scientific basis to carry out measurements and 3D geometric object imaging with the use of coordinate measuring systems [27]. Coordinate measuring machines are designed to measure size, form, and position deviations of a workpiece [22]. A CMM is equipped with a specific type of probing system – contact areal, optical profile, or multi-sensor profile/areal [27]. The probing systems can collect data as single points or a series of points by scanning the workpiece inside the measurement field [22]. Measurement of 3D objects marks the section of the space with the geometry of the measured object given as a point coordinate in a reference coordinate system [27]. Tactile probing systems have roughness limitations due to the stylus tip size diameter. There are many different techniques available for surface measurement analysis and it is important to understand the sample properties, limitations of each, and analysis required [7].

For machining distortion, a coordinate measuring machine may be considered the most universal tool for collecting distortion data, such that it is capable of measuring: a point, line, plane, circle, cylinder, cone, sphere, ellipse, step cylinder, slot, circular slot, parabola, paraboloid, torus, parallel planes, curve, surface, etc. To fully understand the capability of the measurement system, it needs to be analyzed with regard to repeatability, reproducibility, linearity, bias, stability, consistency, and resolution [17]. In the presented study, three industry-relevant coordinate measuring machines at our disposal were chosen with tactile and optical principles.

Table 1 Distortion research examples from the literature often show a vague description of the data collection and analysis methods used.

	[23]	[30]	[18]	[15]	[13]	[2]
Measurement Tool	Taylor Hobson Form Talysurf Scanner	Hexagon Micro HITE 3D CMM	CMM	Mistral 775 3D CMM	CMM	N/A
Measurement Technology	Tactile probe	Tactile probe	N/A	N/A	N/A	N/A
Distortion Visualization	Central line, length direction	Central line, length & width	N/A	3 lines in length equidistant	Grid opposite machined surface	N/A
Distortion Space	Profile	Profile	Area	Profile	Area	Profile
Distortion Analysis	N/A	N/A	N/A	N/A	N/A	N/A

2.2 Distortion characterization

To characterize distortion, workpiece data needs to be measured, processed and visualized. Currently in the literature, distortion measurement includes a variety of methods with regard to measurement tools, tool technology, and distortion space (profile vs. areal measurement) (see Table 1). The analysis methods for the data are often not reported on. Many of the methods used for displaying machining distortion in the literature include a linear or two-dimensional visual representation using a central line for critical distortion maximum areas. In general, distortion measurements and measurands can be described in many ways including:

- a single value (i.e. maximum distortion, average distortion),
- a single curve along a reference surface (usually along the longitudinal direction with maximum deviation noted),
- a collection of curves along a surface (e.g. in both longitudinal and transverse directions with distinct distortion values noted),
- uniformly spaced measurement schemes in both x and y coordinate directions with maximum and minimum values described, etc.

Distortion is often measured as a maximum value or peak–to–valley height on one or several lines across the part. Some researchers take the peak–to–valley height for the whole areal surface. A method for displaying and characterizing machining distortion as areal data with several filtering and fitting steps is described in our previous work [12]. The distortion is calculated from the difference of a reference area measured before and after machining (or pre– and post–). The reference area needs to be largely unaltered by the machining process. Instead of a pre– and post–machining measurement, the post–measurement can be sufficient as long as the reference initial distortion is negligible, i.e., the workpiece surface is flat. Measuring only the post-machining surface for the distortion makes it easier for the operator to collect data and calculate distortion. Also, part quality is defined as difference from the intended shape, which in this case would be the part dimensions as defined by the designer in the CAD drawing, which usually takes an ideal initial condition.

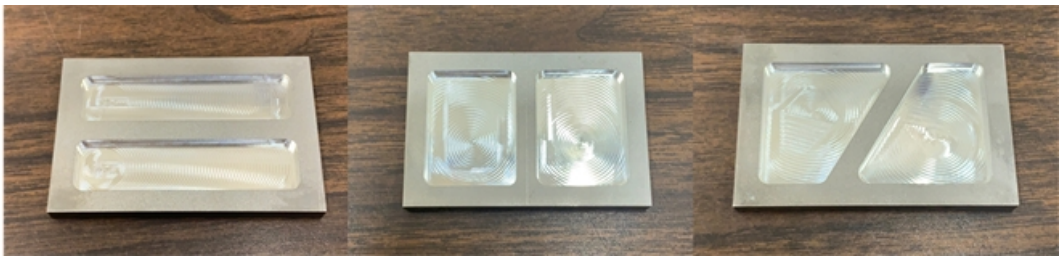


Figure 1 Distortion samples with machined pockets with 0, 90, 60-degrees stiffener walls [14].

Table 2 Specifications for the different CMM machines.

	Tactile CMM	Laser line and tactile probe CMM	Laser triangulation CMM
Model and Manufacturer	Mitutoyo Bright BRT504 CMM	FaroArm Edge & Scan Arm HD	Taylor Hobson Talyscan 250
Accuracy	0.0005 mm	0.029 mm	0.001 mm
Repeatability	0.003 mm	0.025 mm	**
Measurement Technology	Tactile Probe	Laser Line/Tactile Probe	Laser Triangulation
Measuring Capacity	$0.5 \times 0.4 \times 0.4$ m	$1.8 \times 1.8 \times 1.8$ m	$0.2 \times 0.2 \times 0.2$ m

3 Methods

3.1 Workpiece samples

Distortion data was collected on the reference surface (bottom surface) of three aluminum samples with machined pockets from a PAG quenched aluminum 7050 bar. The dimensions of each sample are 76.2 mm x 50.8 mm x 6.35 mm with two equally sized pockets around a stiffener wall at different angles from the sample edge as seen in Figure 1. The stiffener walls are at 0, 90 and 60 degrees. The samples were machined on a 3 axis Haas milling machine after near net shape blanks were cut out by wire-EDM. The pockets were machined with roughing parameters first (4 flute 30 degree helix 1/4" square end mill, trochoidal milling, at 3000 rpm, feed rate of 0.23 m/min, depth of cut of 1.27 mm) and finished with two wall finishing passes to achieve the corner radius (3 flute 30 degree helix 1/8" square end mill, contour milling, at 4500 rpm, feed rate of 0.51 m/min, first depth of cut of 0.38 mm, second depth of cut of 0.127 mm). Flood cooling with a water based coolant was used for all milling operations.

3.2 Coordinate Measuring Machines

Three CMMs were available at our disposal at the University of California Davis for this study: a Mitutoyo Bright BRT 504 CMM (tactile CMM), a Faro Arm Edge and Scan Arm HD CMM (laser line and tactile probe CMM), and a Taylor Hobson Talyscan 250 (laser triangulation CMM) as seen in Figure 2. Initial data collection was carried out on each CMM for investigation of measurement system capabilities and is described in the next section. The specifications and manufacturer technical capabilities of the metrology equipment are presented in Table 2. Note that the repeatability data for the Laser triagulation CMM is unavailable from the manufacturer's specification sheet.

Figure 2 Mitutoyo Bright BRT 504 - tactile CMM (left), Faro Arm Edge and Scan Arm HD - Laser line and tactile probe CMM (middle), and Taylor Hobson Talyscan 250 - Laser triangulation CMM (right).

3.3 Data collection and processing

Data was collected on the bottom surface of each sample, called the reference plane. We chose a single measurement scan instead of pre– and post–machining CMM measurements. The data was collected on each sample using each of the three CMMs for comparison. The comparability of the results was guaranteed because always the whole sample bottom surface was measured. Coordinate data from the CMMs was imported into Matlab software to be fit, leveled and plotted. A planar fit as seen in Equations (1) – (3) was used to level the raw data to the (x, y) plane (Equation (1)). The fit residual (distortion data minus the fitted data) provides leveled data for the following steps. Outliers in the leveled data were filtered further from the leveled data, where outliers are defined as z values near the extremes (maximum and minimum z values). The outliers were removed because they were not part of the measured surface. Instead, they were probing points falling off the edges of the physical part during the measurement. These points are very small percentage of total data points. Defining the z range R as maximum minus minimum (Equation (2)), outliers were identified and removed if outside the range of z from the data by Equation (3). Repeating the planar fit, leveling, and filtering steps three times provided a useful set of data, defined as having no significant outliers and with a difference between fit n and n+1 to be essentially zero. If pre– and post–data sets had been used, the data sets would have to be rotated, translated relative to each other and interpolated linearly to a common grid, before the difference of both data sets can be taken.

$$z(x, y) = a + bx + cy\,, \tag{1}$$

$$R = z_{max} - z_{min}\,, \tag{2}$$

$$z_{\min} + 0.015\,R < z < z_{\max} - 0.015\,R\,. \tag{3}$$

4 Optimizing the measurement parameters

In a first step, measurements were taken on the three above mentioned CMMs. The baseline spacing scheme for the x and y axes on the tactile CMM, and the laser triangulation CMM is 1×1 mm. The spacing scheme for the laser line and tactile probe CMM was found to be about 0.05 x 0.05 mm. Distortion data for each metrology equipment are shown for the

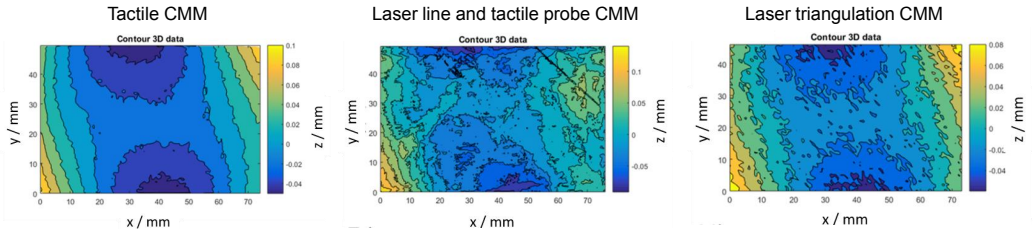


Figure 3 Distortion coordinate data of the sample with 60-degrees stiffener wall from three different CMMs.

60-degree sample in Figure 3. The distortion values (defined as the maximum z–deviations from the intended shape) varied for the different measurement machines and principles. If the scalar distortion values are defined as the maximum distortion value minus the minimum distortion value, the distortion results for the different tools are 0.2438 mm, 0.1504 mm, and 0.1535 mm for the laser line and tactile probe CMM, laser triangulation CMM, and tactile CMM respectively. The laser line and tactile probe CMM has a single point accuracy of 0.029 mm and a volumetric accuracy of ±0.041 mm which contributes to the data noise and to differences from the other coordinate measuring machines. The laser line and tactile probe CMM was therefore determined unfit for the following distortion measurement. The laser triangulation CMM exhibited optical difficulty with the reflective metal surface.

Further investigation of the distortion data from the machined pocket samples was conducted on the tactile CMM. The tactile CMM is often still considered the 'golden standard' in metrology as it represents a physical contact on the workpiece and has the longest history of standardization. The final distortion represents the overall shape features, not the minute texture (i.e. surface roughness).

In a second step, the coordinate spacing for the tactile CMM was investigated to assure quality data with consideration of measurement time. The spacing in the x and y axes were kept uniform. Pitch and increment spacing for the x and y axes respectively were investigated at 5×5 mm, 2×2 mm, 1×1 mm, and 0.5×0.5 mm as seen in Figure 4. Between the 5 x 5 mm and 0.5 x 0.5 mm spacing the general distortion shape does not change. We see that the distortion follows the 60 degrees stiffener wall. But the finer spacing leads to a higher resolution and more information about the surface. However, this must be weighed against the measurement time, which decreases roughly with the square of the spacing distance. Considerable measurement time reduction is possible with the coarser spacing. For the following measurements in this study, a 1 x 1 mm spacing was used. Table 3 shows the available scanning parameters on the chosen measurement device as well as the recommended parameters for the study at hand.

5 Distortion data analysis

The processed distortion data is shown in Figure 5 as 3D plot on the left and line plots for the horizontal midline (y = 25.4 mm) and vertical midline (x = 38.1 mm). The vertical midline shows a convex shape, the horizontal midline a concave shape.

As a new data analysis step, the leveled data is also imported into a least–squares surface fitting program that creates a Bézier surface, whose shape may be characterized by analysis of its Gaussian curvature. An illustration of the coordinate data analysis steps are shown in Figure 6.

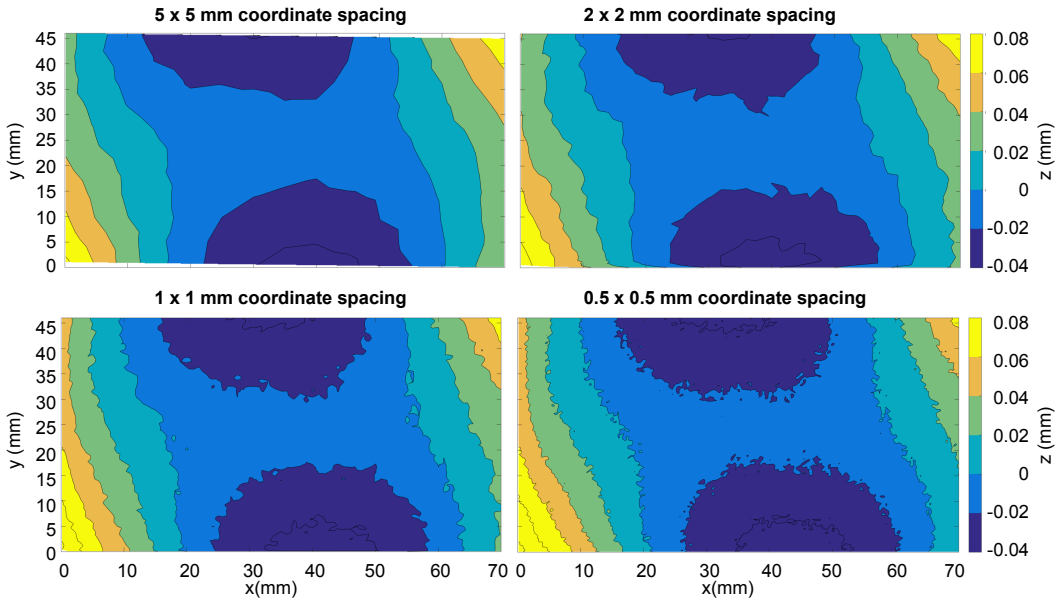


Figure 4 Distortion coordinate data of the sample with 60-degrees stiffener wall from the tactile CMM using different coordinate spacing schemes.

Table 3 Tactile CMM distortion measurement parameters.

CMM Scanning Parameter	**Definition or available options**	**Recommended Parameters**
x-spacing	Increment spacing between points in the x direction	1 mm
y-spacing	Pitch spacing between points in the y direction	1 mm
Lace feature	Repeat scan direction uniform or positive/negative direction	off
Scan method	Peck or Drag	Peck
Scan Transversal	x, y or z-axis	y-axis
Pecking axis	x, y, or z-axis	z-axis
Pecking direction	Positive or negative axis (x,y,z) direction	Negative z-axis
No. of T-patches	Length of the part / pitch	25
Retreat axis	x, y or z-axis	z-axis
Probe diameter	Size of the scanning probe ball diameter	4 mm
Probe Compensation	Coordinate datum is compensated using the radius of the scanning probe	On

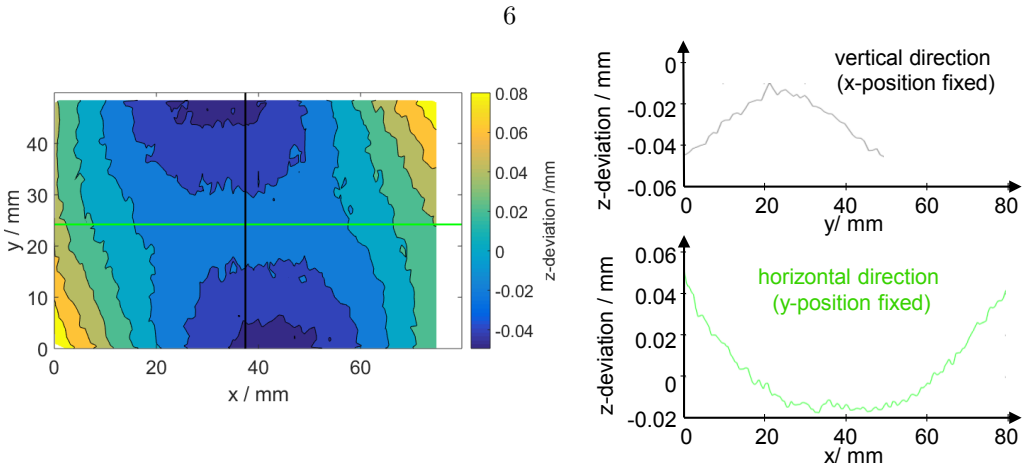


Figure 5 The leveled distortion data for the 60-degree sample with midline plots for both horizontal and vertical directions.

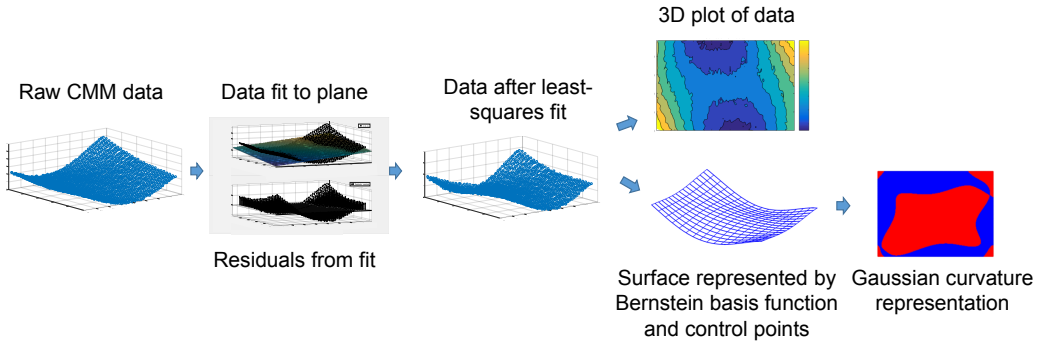


Figure 6 Steps for coordinate distortion data processing and visual representation.

After the data has been leveled, it is reduced to a uniform 500×500 2D grid using *meshgrid* and *griddata* in Matlab. The distortion is displayed as a contour plot. After leveling the data in Matlab, a least-squares method is used to remove any noise in the original coordinate data. The idea is to construct a Bézier surface with a grid of control points. The least-squares problem reduces to a linear system that is relatively easy to solve. The height distributions will give an idea about the local distortion shape (convex, saddle, etc.). The algorithmic steps for this optimized routine for characterizing machining distortion are presented below.

5.1 Least-Squares Fitting with Bézier Surfaces

Since the original leveled CMM coordinate data is noisy, it can be difficult to discern overall shape properties characterizing the distortion directly from it. To address this, we consider least–squares fitting of the data to polynomial Bézier surfaces to suppress the noise. The least–squares fit involves only the solution of a linear system of equations, and allows two–dimensional shape features of the data to be identified, rather than just properties along linear subsets (as with conventional methods).

The Bernstein polynomial basis of degree n over the interval $t \in [\,0,1\,]$ is defined by

$$b_k^n(t) \;=\; \binom{n}{k}(1-t)^{n-k}t^k\,, \quad k=0,\ldots,n\,,$$

and a polynomial $p(t)$ may be specified in terms of its Bernstein coefficients $c_0,\ldots,c_n$ through the expression

$$p(t) \;=\; \sum_{k=0}^{n} c_k b_k^n(t)\,.$$

The Bernstein basis has many advantageous properties [9] in the context of least–squares fitting of polynomials to noisy data over finite domains:

1. *unimodality*: $b_k^n(t)$ has a single maximum on $t \in [\,0,1\,]$;
2. *non–negativity*: $b_k^n(t) \geq 0$ on $t \in [\,0,1\,]$ for $k=0,\ldots,n$;
3. *partition of unity property*: $\sum_{k=0}^{n} b_k^n(t) = 1$ for all values of t;
4. *lower and upper bounds*: $\min_{0\leq k\leq n} c_k \leq p(t) \leq \max_{0\leq k\leq n} c_k$ for $t \in [\,0,1\,]$;
4. *variation–diminishing property*: the number R of real roots of $p(t)$ on $t \in (0,1)$ is less than the number $V(c_0,\ldots,c_n)$ of sign variations in its Bernstein coefficients by an even amount, i.e., $R = V(c_0,\ldots,c_n) - 2\,K$ for a non–negative integer K;
5. *differentiation and integration*: the derivative and the integral of $p(t)$ can be expressed as polynomials in Bernstein form of degree $n-1$ and $n+1$, respectively, with coefficients that are simple linear combinations of the coefficients $c_0,\ldots,c_n$;
6. *numerical stability*: the Bernstein basis is "optimally stable" among all non–negative polynomial bases on $[\,0,1\,]$ – i.e., it is impossible to construct a basis for which the values of $p(t)$ are systematically less sensitive to uniform relative perturbations of its coefficients.

A tensor–product Bézier surface $\mathbf{r}(u,v)$ of degree (d,d) on the domain $(u,v) \in [\,0,1\,]\times[\,0,1\,]$ is defined in terms of the Bernstein bases in the parameters u and v through its *control points* $\mathbf{p}_{kl}$ for $0 \leq k,l \leq d$ by the expression

$$\mathbf{r}(u,v) \;=\; \sum_{k=0}^{d}\sum_{l=0}^{d} \mathbf{p}_{kl}\, b_k^d(u) b_l^d(v)\,. \tag{4}$$

The least–squares surface fit of the leveled CMM coordinate data proceeds as follows. We are given the measured heights z_i at sample points (x_i,y_i) for $1 \leq i \leq N$ on the domain $(x,y) \in [\,x_{\min},x_{\max}\,] \times [\,y_{\min},y_{\max}\,]$. The Bernstein basis functions of degree d in x and y on $[\,x_{\min},x_{\max}\,]$ and $[\,y_{\min},y_{\max}\,]$ are defined by

$$b_k^d(x) \;=\; \binom{d}{k}\frac{(x_{\max}-x)^{d-k}(x-x_{\min})^k}{(\Delta x)^d}\,, \quad k=0,\ldots,d\,,$$

$$b_l^d(y) \;=\; \binom{d}{l}\frac{(y_{\max}-y)^{d-l}(y-y_{\min})^l}{(\Delta y)^d}\,, \quad l=0,\ldots,d\,,$$

where $\Delta x = x_{\max}-x_{\min}$, $\Delta y = y_{\max}-y_{\min}$. We wish to fit a (functional) tensor–product surface of degree (d,d) in (x,y) of the form

$$z \;=\; f(x,y) \;=\; \sum_{k=0}^{d}\sum_{l=0}^{d} c_{kl}\, b_k^d(x) b_l^d(y) \tag{5}$$

to the given data. The total squared error between the fitted surface and the given data is defined as

$$E \;=\; \sum_{i=1}^{N} \left[\, f(x_i, y_i) - z_i \,\right]^2 ,$$

and substituting from (equation 5) gives

$$E \;=\; \sum_{i=1}^{N} \left[\sum_{k=0}^{d} \sum_{l=0}^{d} c_{kl}\, b_k^d(x_i) b_l^d(y_i) - z_i \right]^2 .$$

The least–squares fit is obtained by minimizing the error E with respect to the coefficients c_{kl} of $f(x, y)$. Setting the derivative of E with respect to the coefficient c_{rs} equal to zero yields the equation

$$\frac{\partial E}{\partial c_{rs}} \;=\; \sum_{i=1}^{N} 2 \left[\sum_{k=0}^{d} \sum_{l=0}^{d} c_{kl}\, b_k^d(x_i) b_l^d(y_i) - z_i \right] b_r^d(x_i) b_s^d(y_i) \;=\; 0 \,,$$

or equivalently,

$$\sum_{k=0}^{d} \sum_{l=0}^{d} \left[\sum_{i=1}^{N} b_k^d(x_i) b_l^d(y_i) b_r^d(x_i) b_s^d(y_i) \right] c_{kl} \;=\; \sum_{i=1}^{N} b_r^d(x_i) b_s^d(y_i)\, z_i \,.$$

For each pair (r, s) with $0 \le r, s \le d$ this defines a system of $(d+1)^2$ linear equations for the unknown coefficients c_{kl} of the surface (Equation 5).

The linear equations can be solved by standard methods (e.g., Gaussian elimination). To do this, it is preferable to express the equations in standard matrix form. This is accomplished by setting

$$\mu = r(d+1) + s + 1 \quad \text{and} \quad \nu = k(d+1) + l + 1$$

for $0 \le r, s \le d$ and $0 \le k, l \le d$, so the linear equations can be written as

$$\sum_{\nu=1}^{(d+1)^2} M_{\mu\nu}\, \tilde{c}_\nu \;=\; r_\mu \,, \quad \mu = 1, \ldots, (d+1)^2$$

where the matrix elements, unknowns, and right–hand side values are

$$M_{\mu\nu} \;=\; \sum_{i=1}^{N} b_r^d(x_i) b_s^d(y_i) b_k^d(x_i) b_l^d(y_i) \,, \quad 1 \le \mu, \nu \le (d+1)^2 \,,$$

$$\tilde{c}_\nu \;=\; c_{kl} \,, \quad 1 \le \nu \le (d+1)^2 \,,$$

$$r_\mu \;=\; \sum_{i=1}^{N} b_r^d(x_i) b_s^d(y_i)\, z_i \,, \quad 1 \le \mu \le (d+1)^2 \,.$$

Once the solution vector $\tilde{c}_\nu$ for $\nu = 1, \ldots, (d+1)^2$ has been computed, it can be re–arranged as the two–dimensional array c_{kl} with $0 \le k, l \le d$ that defines the surface (Equation 5) by writing

$$c_{kl} \;=\; \tilde{c}_\nu \,, \quad \text{where} \;\; k = \lfloor (\nu - 1)/(d+1) \rfloor \,, \quad l = \nu - k(d+1) - 1 \,.$$

5.2 Surface curvature analysis

To analyze the shape of the surface with Equation (4) we must rely on *intrinsic* properties – i.e., shape measures that are independent of the surface parameterization. The *Gaussian curvature* is the most basic intrinsic measure of local surface shape. At every surface point, there exists a family of planes that contain the surface normal vector, each of which intersects the surface in a planar *normal section curve*. The curvature of each normal section curve specifies a *normal curvature* of the surface at a given point, and there are (in general) two mutually orthogonal orientations of the section planes for which the normal curvature is extremal. These section planes identify the *principal directions* and *principal curvatures* of the surface at each point, and the Gaussian curvature at each point is the product of the principal curvatures.

At a given point, a surface is "bowl–shaped" or "saddle–shaped" according to whether the Gaussian curvature at that point is positive or negative. For example, an ellipsoid has positive Gausssian curvature at every point, but a hyperboloid has negative Gaussian curvature at every point. A general "free–form" surface may exhibit regions of both positive and negative Gaussian curvature, separated by loci of zero Gaussian curvature called *parabolic lines*. Analysis of the Gaussian curvature of the distorted surfaces produced by machining can reveal whether they admit simple characterizations, or are inherently rather complex.

The coefficients of the *first fundamental form* of a parametric surface $\mathbf{r}(u, v)$ are defined in terms of its first partial derivatives as

$$E \,=\, \mathbf{r}_u \cdot \mathbf{r}_u \,, \quad F \,=\, \mathbf{r}_u \cdot \mathbf{r}_v \,, \quad G \,=\, \mathbf{r}_v \cdot \mathbf{r}_v \,.$$

The *second fundamental form* has coefficients given in terms of the surface normal vector

$$\mathbf{n} \,=\, \frac{\mathbf{r}_u \times \mathbf{r}_v}{|\, \mathbf{r}_u \times \mathbf{r}_v \,|}$$

and the surface second partial derivatives by

$$L \,=\, \mathbf{n} \cdot \mathbf{r}_{uu} \,, \quad M \,=\, \mathbf{n} \cdot \mathbf{r}_{uv} \,, \quad N \,=\, \mathbf{n} \cdot \mathbf{r}_{vv} \,.$$

The Gaussian curvature is determined in terms of the coefficients of the first and second fundamental forms as

$$K \,=\, \frac{LN - M^2}{EG - F^2} \,.$$

Hence, a neighborhhod of a point is "bowl–shaped" or "saddle–shaped" according to whether $K > 0$ or $K < 0$, while $K = 0$ identifies points on a parabolic lines (a surface with $K \equiv 0$ is a *developable* surface – imagined as a thin material sheet, it can be "flattened" onto a plane without stretching or compressing the material).

For the case of a "functional surface" specified as $z = f(x, y)$ we may take $u = x$, $v = y$ and the parameterization has the form $\mathbf{r}(x, y) = (x, y, f(x, y))$ with the properties

$$\mathbf{r}_x \,=\, (1, 0, f_x) \,, \quad \mathbf{r}_y \,=\, (0, 1, f_y) \,, \quad \mathbf{n} \,=\, \frac{(-f_x, -f_y, 1)}{\sqrt{f_x^2 + f_y^2 + 1}} \,,$$

$$\mathbf{r}_{xx} \,=\, (0, 0, f_{xx}) \,, \quad \mathbf{r}_{xy} \,=\, (0, 0, f_{xy}) \,, \quad \mathbf{r}_{yy} \,=\, (0, 0, f_{yy}) \,,$$

$$(E, F, G) \,=\, (f_x^2 + 1, f_x f_y, f_y^2 + 1) \,, \quad (L, M, N) \,=\, \frac{(f_{xx}, f_{xy}, f_{yy})}{\sqrt{f_x^2 + f_y^2 + 1}} \,,$$

$$K = \frac{f_{xx} f_{yy} - f_{xy}^2}{(f_x^2 + f_y^2 + 1)^2} \,. \tag{6}$$

After successfully implementing the optimized routine for characterizing machining distortion via Gaussian surface curvature, the results are displayed below.

6 Results and Discussion

The leveled data shown in Figure 5 depicts more information in the left contour plot, than in the two midline plots on the right. The distortion seen in the 60-degree sample is symmetric and complex; both of these properties are not evident from the line plots. Areal data yields more information for the user than profile data. But still the surface shape cannot easily be described from the depicted 3D plot.

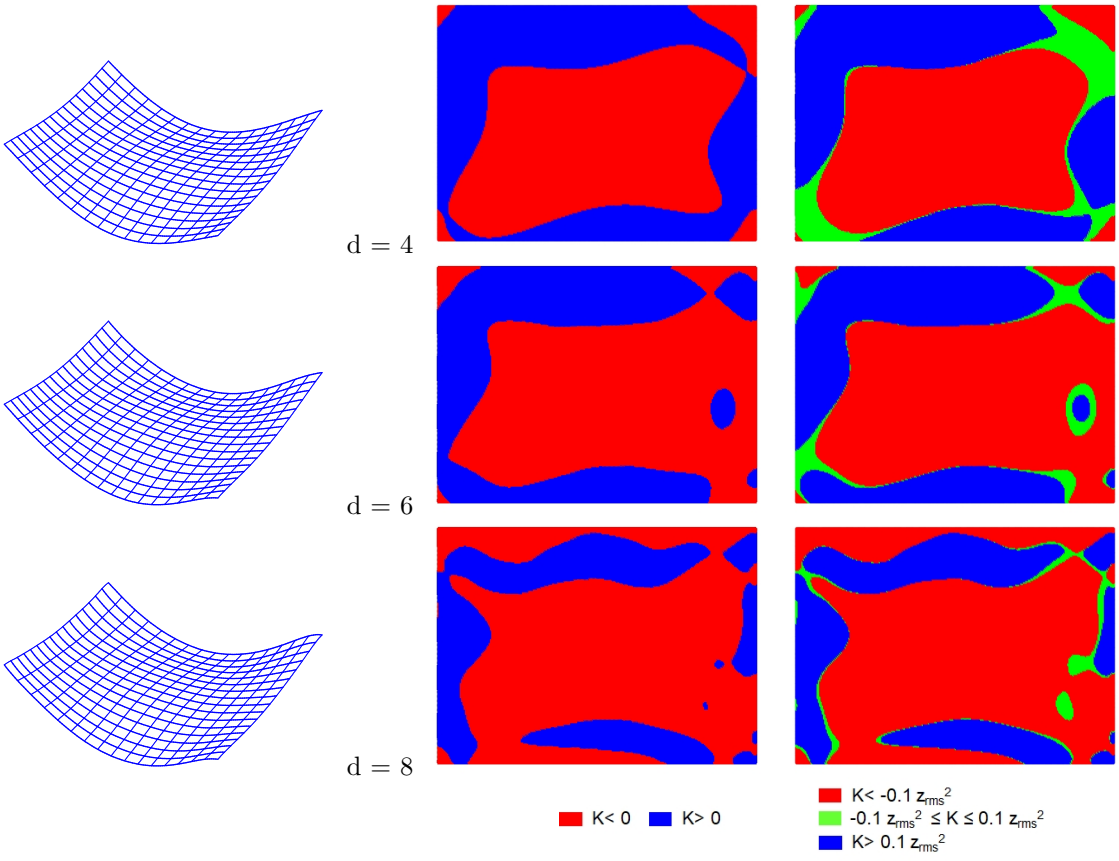


Figure 7 Least–squares surface fits with degrees $d = 4$ (top), $d = 6$ (center), and $d = 8$ (bottom). Left: height deviation of least–squares surface about the mean height (magnified 120×). Center: surface regions with negative (red) and positive (blue) Gaussian curvature K. Right: surface regions that satisfy $K > 0.1\, z_{\text{rms}}^2$ (blue), $-0.1\, z_{\text{rms}}^2 \le K \le 0.1\, z_{\text{rms}}^2$ (green), and $K < -0.1\, z_{\text{rms}}^2$ (red).

The leveled data was also used to compute a least–squares surface fit. From the leveled data of $N = 3825$ data points, the Gaussian curvature Equation (6) of the fitted surface was computed. Figure 7 shows the regions of positive and negative Gaussian curvature of the

fitted surface Equation (5) with degrees $d = 4, 6, 8$. Because the data points represent only a very mild deviation from planarity, the identification of these regions is sensitive to the number $(d+1)^2$ of fitting parameters – as d is increased, the fitted surface begins to more accurately replicate the noise inherent in the measured data.

Figure 7 also gives an alternative view, in which regions of low Gaussian curvature magnitude $|K| < 0.1\, z_{\text{ms}}^2$ (where z_{rms} is the root–mean–square deviation of the z heights) are color–coded green, while regions with $K > 0.1\, z_{\text{ms}}^2$ and $K < -0.1\, z_{\text{ms}}^2$ are colored red and blue. The green regions are nearly developable, indicating that the surface shape results from "bending" (rather than stretching or compressing) of an initially flat surface.

Figure 8 shows the magnified surface deviations and Gaussian curvature variations for the three test parts illustrated in Figure 1. A strong correlation of the curvature with the orientation of the stiffening wall is apparent. Overall, Figures 7 and 8 suggest that the surface distortions produced by machining are rather complex in nature, and do not admit characterization by simple, intuitive shape parameters.

Using the Gaussian curvature in addition to the areal distortion representation provides greater insight into the the overall surface deviation in the context of distortion analysis. Lower degrees d of the surface in the x and y directions impart a simpler impression of the overall distorted surface shape. As d increases, more features related to surface topography (i.e. surface roughness) are emphasized. Reducing the degree d incurs a smoothing effect that emphasizes the broad surface shape: as $(d+1)^2$ approaches N (the number of data points), the least–squares fit attempts to exactly interpolate every data point, yielding a Gaussian curvature distribution that is noisy, reflecting the surface roughness. Distortion data presented as either as a line or curve only exhibits a fraction of the overall experienced distortion on that sample. Two–dimensional data gives a sparse indication of the shape and does not adequately represent the complexity of the distorted shape.

7 Conclusion

This study explains an updated routine for characterizing machining distortion to improve comparison and transfer of distortion research between different studies. The optimized routine for characterizing machining distortion can be seen as a process to investigate the complexities of the unique distorted surface. The overall characterization process as seen in Figure 6 shows the steps to analyze distortion with Bézier surfaces and Gaussian curvature. Different representations of distortion will present more information about the distortion than just a single line plot or a single 3D image and help to understand machining distortion better.

Through this research, it is understood that distortion often includes complex shapes with regions of the surface between positive and negative Gaussian curvature. A new transparent way to characterize distortion via Gaussian curvature proves to be a simple method to transfer distortion results. Using the Bernstein basis function as a means for the least-squares surface fit for Gaussian curvature extraction of distorted machined samples, expresses a novel method to characterize machining distortion for the overall complex shape deviation. Extracting information from the curvature including symmetries, quantified positive and negative distributions, curvature location with respect to machined geometry, etc., can be useful for further distortion characterization and minimization.

More work is needed using Gaussian curvature characterization to expand the distortion results to show surface curvature symmetries, curvature percentages of both positive and negative curvatures, and distinct characterization Gaussian shapes for different distorted

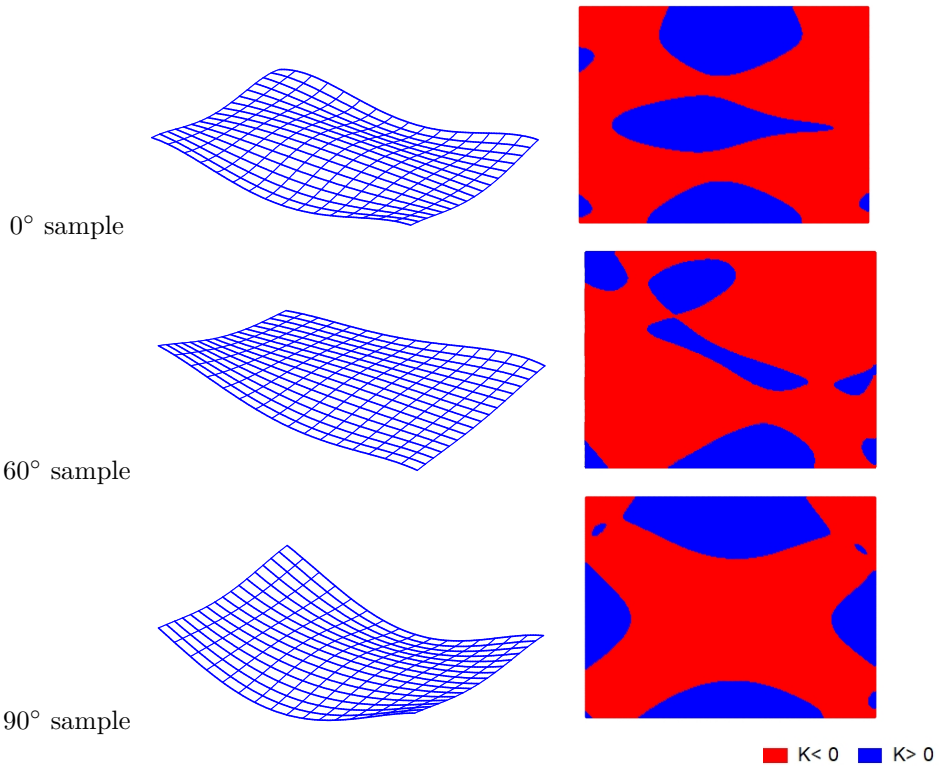


Figure 8 Left: height deviation of least–squares surface fit from the mean height (magnified 120×) for degree $d = 4$ surfaces. Right: plots showing regions of negative (red) and positive (blue) Gaussian curvature. The upper case shows the 0° *sample*; the middle case is the 60° *diagonal sample*; and the lower case is the 90° *sample*.

parts. Further work is also necessary in distortion compensation and distortion control. For example, Finite Element Analysis of distortion from residual stresses can use Gaussian Curvature to describe the distorted surfaces and can then be compared to Gaussian Curvature data from measured surfaces. Using the final distorted curvatures can allow for the prediction and manipulation of future machining distortion minimization. Future work is also needed to investigate more geometries and dimensions (i.e. complex geometries and pockets).

References

1 S. Bell. *Measurement Good Practice Guide No. 11. A Beginner's Guide to Uncertainty of Measurement.* Tech. rep., National Physical Laboratory, 1999.

2 Yun-bo Bi, Qun-lin Cheng, Hui-yue Dong, and Ying-lin Ke. Machining distortion prediction of aerospace monolithic components. *Journal of Zhejiang University-SCIENCE A*, 2009. doi:10.1631/jzus.a0820392.

3 DM Bowden and JE Halley. *Aluminum reliability improvement program final report 60606.* The Boeing Company, Chicago, IL, USA, 2001.

4 E. Brinksmeier, J.T. Cammett, W. König, P. Leskovar, J. Peters, and H.K. Tönshoff. Residual stresses — measurement and causes in machining processes. *CIRP Annals*, 31(2):491–510, 1982. doi:10.1016/S0007-8506(07)60172-3.

5 E. Brinksmeier and Jens Sölter. Prediction of shape deviations in machining. *CIRP Annals*, 58:507–510, December 2009. doi:10.1016/j.cirp.2009.03.123.

6 D. Chantzis, S. Van der Veen, J.Zettler, and W.M. Sim. An industrial workflow to minimise part distortion for machining of large monolithic components in aerospace industry. *Procedia CIRP*, 8:281–286, 2013. 14th CIRP Conference on Modeling of Machining Operations (CIRP CMMO). doi:10.1016/j.procir.2013.06.103.

7 M Conroy and J Armstrong. A comparison of surface metrology techniques. *Journal of Physics: Conference Series*, 13:458–465, 2005.

8 Richard Leach David Flack, James Claverley. Chapter 9 - coordinate metrology. In Richard Leach, editor, *Fundamental Principles of Engineering Nanometrology (Second Edition)*, Micro and Nano Technologies, pages 295–325, Oxford, 2014. William Andrew Publishing. doi:10.1016/B978-1-4557-7753-2.00009-8.

9 Rida T. Farouki. The bernstein polynomial basis: A centennial retrospective. *Computer Aided Geometric Design*, 29(6):379 – 419, 2012. doi:10.1016/j.cagd.2012.03.001.

10 Joint Committee for Guides in Metrology. *JCGM 104:2009. Evaluation of measurement data – An introduction to the "Guide to the expression of uncertainty in measurement" and related documents*, volume First edition. JCGM, 2009.

11 International Organization for Standardization. ISO 10360-1:2000 geometrical product specifications (gps) — acceptance and reverification tests for coordinate measuring machines (cmm) — part 1: Vocabulary). *Vernier, Geneva, Switzerland*, 2000.

12 Destiny R. Garcia, Michael R. Hill, Jan C. Aurich, and Barbara S. Linke. Characterization of machining distortion due to residual stresses in quenched aluminum. *Proceedings of the ASME 2017 12th International Manufacturing Science and Engineering Conference collocated with the JSME/ASME 2017 6th International Conference on Materials and Processing*, Volume 1: Processes, June 2017. V001T02A031. doi:10.1115/MSEC2017-2878.

13 M. Gulpak, J. Sölter, and E. Brinksmeier. Prediction of shape deviations in face milling of steel. In *Procedia CIRP*, 2013. doi:10.1016/j.procir.2013.06.058.

14 Michael Hill, Christopher D'Elia, Renan Ribeiro, and Julianne Emily Jonsson. Measurement and prediction of distortion from bulk residual stress in aluminum 7050 coupons (in preparation). *TBD*, 2021.

15 Xiaoming Huang, Jie Sun, and Jianfeng Li. Finite element simulation and experimental investigation on the residual stress-related monolithic component deformation. *International Journal of Advanced Manufacturing Technology*, 2015. doi:10.1007/s00170-014-6533-9.

16 Rémi Husson, Jean-Yves Dantan, Cyrille Baudouin, Serge Silvani, Thomas Scheer, and Régis Bigot. Evaluation of process causes and influences of residual stress on gear distortion. *CIRP Annals*, 61(1):551–554, 2012. doi:10.1016/j.cirp.2012.03.106.

17 ASTM International. ASTM E2782-17 standard guide for measurement systems analysis (msa). *West Conshohocken, PA*, 2017. doi:10.1520/E2782-17.

18 S. Jayanti, D. Ren, E. Erickson, Shuji Usui, T. Marusich, K. Marusich, and H. Elanvogan. Predictive modeling for tool deflection and part distortion of large machined components. *Procedia CIRP*, 12:37–42, December 2013. doi:10.1016/j.procir.2013.09.008.

19 Fritz Klocke. *Manufacturing Processes 1*. Springer Berlin Heidelberg, 2011. doi:10.1007/978-3-642-11979-8.

20 Ismail Lazoglu, Durul Ulutan, B. Alaca, Serafettin Engin, and Bilgin Kaftanoğlu. An enhanced analytical model for residual stress prediction in machining. *CIRP Annals - Manufacturing Technology*, 57:81–84, December 2008. doi:10.1016/j.cirp.2008.03.060.

21 Jian-guang Li and Shu-qi Wang. Distortion caused by residual stresses in machining aeronautical aluminum alloy parts: recent advances. *The International Journal of Advanced Manufacturing Technology*, 89:997–1012, March 2017. doi:10.1007/s00170-016-9066-6.

22 Karsten Luebke. Coordinate measuring machine. In Luc Laperrière and Gunther Reinhart, editors, *CIRP Encyclopedia of Production Engineering*, pages 285–289, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg. doi:10.1007/978-3-642-20617-7_6579.

23 Ninshu Ma and Hui Huang. Efficient simulation of welding distortion in large structures and its reduction by jig constraints. *Journal of Materials Engineering and Performance*, 26:5206–5216, 2017.

24 Daniel Jon Mitchell, Eran Tal, and Hasok Chang. The making of measurement: Editors' introduction. *Studies in History and Philosophy of Science Part A*, 65-66:1–7, 2017. The Making of Measurement. doi:10.1016/j.shpsa.2017.10.001.

25 Enrico Savio, L. Chiffre, Simone Carmignato, and J. Meinertz. Economic benefits of metrology in manufacturing. *CIRP Annals - Manufacturing Technology*, 65:495–498, January 2016. doi:10.1016/j.cirp.2016.04.020.

26 Wei-Ming Sim. Challenges of residual stress and part distortion in the civil airframe industry. *International Journal of Microstructure and Materials Properties*, 5, December 2010. doi:10.1504/IJMMP.2010.037621.

27 Jerzy A. Sładek. *Coordinate Metrology*. Springer Berlin Heidelberg, 2016. doi:10.1007/978-3-662-48465-4.

28 Klaus-Dieter Thoben, Thomas Lübben, Brigitte Clausen Christian Prinz, Alwin Schulz, Rüdiger Rentsch, Ralf Kusmierz, Lutz Nowag, Holger Surm, Friedhelm Frerichs, Martin Hunkel, Dieter Klein, and Peter May. Distortion Engineering: Eine systemorientierte Betrachtung des Bauteilverzugs. *HTM - Haerterei-Technische Mitteilungen*, 57:276–282, April 2002.

29 PF FEng Y Ma, S Liu and DW Yu. Finite element analysis of residual stresses and thin plate distortion after face milling. In *12th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad*, pages 67–71, 2015. doi:10.1109/IBCAST.2015.7058481.

30 Zheng Zhang, Liang Li, Yinfei Yang, Ning He, and Wei Zhao. Machining distortion minimization for the manufacturing of aeronautical structure. *International Journal of Advanced Manufacturing Technology*, 2014. doi:10.1007/s00170-014-5994-1.

Interactive Quality Inspection of Measured Deviations in Sheet Metal Assemblies

Felix Claus[1]
Computer Graphics & HCI Lab, Computer Science Department,
Technische Universität Kaiserslautern, Germany

Hans Hagen
Computer Graphics & HCI Lab, Computer Science Department,
Technische Universität Kaiserslautern, Germany

Viktor Leonhardt
Scientific Visualization Lab, Computer Science Department,
Technische Universität Kaiserslautern, Germany

Heike Leitte
Visual Information Analysis, Computer Science Department,
Technische Universität Kaiserslautern, Germany

Bernd Hamann
Department of Computer Science, University of California, Davis, CA 95616, USA

Abstract

We present an exploratory data analysis approach for finite element (FE) simulations to interactively inspect measured deviations in sheet metals arising in automotive applications. Exterior car body parts consist of large visible surfaces, and strict tolerances must be met by them to satisfy both aesthetic requirements and quality performance requirements. To fulfill quality requirements like gap and flushness, exterior vehicle components have adjustable mechanical boundaries. These boundaries are used to influence the shape and position of a sheet metal part relative to its chassis. We introduce a method that supports an inspection engineer with an interactive framework that makes possible a detailed analysis of measured sheet metal deviation fields generated from 3D scans. An engineer can interactively change boundary conditions and obtains the resulting deviation field in real-time. Thus, it is possible to determine viable and desirable adjustments efficiently, leading to time and cost savings in the assembly process.

2012 ACM Subject Classification Applied computing

Keywords and phrases Data Analysis, Interactive Inspection, 3D-Metrology, Finite Element Simulation

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.6

Funding This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057.

Acknowledgements We thank Q-DAS GmbH for providing CAD models.

1 Introduction

Measuring the dimensions of produced car body parts is one of the most important aspects of quality assurance. Regarding sheet metal assembly processes in today's automotive industry, dimension assessment is usually condensed by measuring single points on an assembled part. These arbitrary points, referred to as KPIs (key performance indicators) must be chosen

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 6; pp. 6:1–6:18

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

carefully to obtain a satisfactory assessment of the quality of a measured part. We focus on the assembly of exterior automotive chassis components. KPIs are chosen such that gap and flushness between parts can be controlled. These properties have small tolerance ranges, and it is difficult to fine-tune them. The chosen tolerances are usually below the dimensional variability of the assembled parts. Adjustable boundaries are used to fine-tune gap and flushness for every assembly. When using the term "boundary" in the following, we refer to an actual physical boundary of a mechanical part. To support adaptive assembly, expensive tools with mounted measurement devices are employed to control gaps and flushness. However, additional inspection steps are needed to evaluate an assembly produced in an automated process. During detailed analysis, more inspection points are evaluated than can be actively controlled during the assembly process. To increase the amount of information captured by

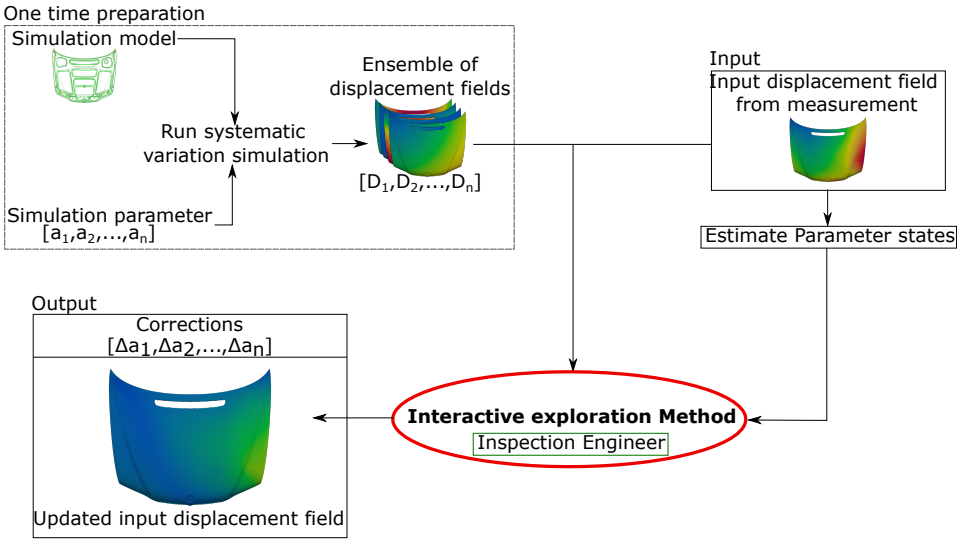


Figure 1 High-level description of workflow outline, grouped by the following tasks/data: "One-time preparation," "Input" and "Output".

a single measurement, optical measurement methods have become increasingly important for assessing sheet metal assemblies. Optical measurement technology captures the entire geometry of the visible surface of a produced assembly at high resolution. The "point clouds" generated by scanning can provide more detailed information about a measured geometry but, at the same time, it can be difficult to interpret the scanned data. Scans can be difficult to understand since a comparison between desired geometry and measured data results in 3D deviation (or difference) vector fields. To support simpler visual interpretation, we plot signed distance to the geometry instead of rendering the 3D offset vector field. When inspecting these color-coded distance fields, effects like dents, buckling, or elevations of large areas can be identified. However, the reason for identified effects can be a superposition of misplaced boundaries. Presumed geometrical compliance of the individual assembly components is already taken into account. The determination of appropriate countermeasures for greatly reducing identified deviations requires an experienced quality engineer with a substantial understanding of part behavior. Especially when parts are used for the first time it is often necessary to invest significant time to obtain the needed knowledge.

To speed up the mentioned trial and error experiments, this paper proposes an explorative approach to interact with the measured deviations. Figure 1 shows a high-level description of the workflow and the integration of the method. The figure is separated into "one-time

preparation", "Input" and "Output". As one-time preparation steps, an FE-simulation model must be set up from CAD geometry and an ensemble of displacement fields must be simulated sampling the solution space spanned by the simulation parameters that are considered adjustable. As input for the exploration method, a deviation field is calculated by comparing a measurement of an assembled part with its desired geometry. The obtained deviation field is then analyzed by a parameter estimation method to find a similar parameter state in the ensemble of pre-calculated displacement fields. Based on the obtained parameter state the input displacement field can then be updated by interpolating in between the pre-calculated displacement fields. So the user can change the parameters and gets a real-time update of the measurement. When the user found a satisfactory result, the output of the method is an updated displacement field and a set of necessary corrections. To meet the requirements of a system that is capable of updating measurements in real-time, our goal is to compute the update of the measurement below $0.5s$.

2 Background

This section summarizes the state of the art affecting the method this paper proposes.

Causal analysis of sheet metal assembly errors

The identification of potential causes for measured deviations is essential for quality assurance. Once the reason(s) is(are) identified, countermeasures can be executed. The concept of causal analysis is related to the topic of this paper, as the aim is to interactively identify proper countermeasures for measured deviations. [7] uses this concept for reducing the dimensional variation in multi-stage assemblies. When performing a causal analysis usually all possible influences on the process are considered and explanations for variation are identified. Typical causal factors in sheet metal assembly are, for example, assembly tool variation, part variation, the order of fastening screws or spot welds. Published approaches deal with optimizing these identified weak spots, see [17], [6], [16], [5], [15].

Parameter Estimation

Parameter estimation is widely used in different applications for identifying unknown quantities or states in simulation models, see [26], [3], [22]. In FE simulation, parameter estimation can be used for fine-tuning material parameters of a simulation model, feeding the estimator with experimental data, see [11] and [25]. [20] use parameter estimation as a kernel of a closed-loop control system for automotive purposes to predict critical motion states. The most relevant branch of parameter estimation methods used for our work is based on parameter estimation by least-squares methods. An overview of these methods is given in [13]. We already set up and validated a parameter estimation approach utilizing a least-squares method in connection with 3D-scanned deviations in sheet metal assemblies, presented in [8].

Proper Orthogonal Decomposition (POD)

POD is a branch of dimension reduction methods for finding a lower-dimensional basis for high-dimensional data. A review of POD methods is given in [18]. PODs are used in a wide variety of applications. For example, [4] use decomposition for reconstructing pressure fields on aerofoils from measured point data. [2] use POD for reducing the amount of necessary

single optical measures for recognizing a sheet metal part by reconstructing non-measured areas. The principle of POD is used in this paper to solve the parameter estimation problem for 3D-vector field applications.

Interactive/Explorative Design

Interactive design methods are commonly used for finding an optimal shape or design of a product to fulfill defined specifications by exploring the design space. These methods are used for a wide variety of applications and domains, see [21], [19], [24] . Especially in engineering applications, interactive design methods in combination with physical simulations become more popular, see [14], [9]. The method proposed in this paper is a hybrid approach containing aspects from interactive design methods as well as aspects from data exploration. To briefly cover literature from the field of data exploration we refer to an overview given in [12].

The presented literature covers many diverse topics of research spread over different disciplines. This work utilizes the knowledge from these different disciplines combining it to create a novel approach for interacting with measured sheet metal deviations. In particular, the contributions are:

- Method for supporting decision-making processes in the quality inspection of sheet metal assemblies
- Applicable for real-time interaction
- Validations performed with simulation- and experimental data

This paper unfolds as follows. First, the method is described in Section 3 covering different aspects like preparation steps and data acquisition, as well as a mathematical and algorithmic description of the method. Next, Section 4 presents a use case scenario showing validations with simulation- and experimental data. Last, the results are discussed in Section 5 and the paper is concluded by Section 6.

3 Method

In this section, the method that enables the interaction with the measurement data is explained in detail. Figure 2 depicts a detailed workflow that focuses on the “Interactive Exploration Method” mentioned in Figure 1 (red ellipse). In Figure 2 the interactive exploration method is also outlined with a red box. As input, we need an ensemble of pre-simulated displacement fields and the used simulation parameters. Also, the parameter states that led to the scanned deviation must be identified by a parameter estimation method. The estimated parameter states are compared to the pre-simulated parameters to determine the position of the scanned deviation field relative to the sampled points in the parameter space. Next, the deviation field can be represented by an interpolation between the pre-simulated displacement fields. This is achieved by weighting the pre-simulated fields corresponding to the distance calculated in the parameter space. At this point in the workflow, the user can make an input by changing the initially estimated parameter states of the scanned deviation. This kicks off an update of the calculated distances and thus the weights for the pre-simulated displacement fields become recalculated as well. By that, the scanned deviation field can be updated by the change of the new interpolation result. This update can be performed within a fraction of a second. So the user can interactively try out changes of parameters until he achieves a satisfying result. The loop depicted in Figure 2 is representing the update loop due to user input.

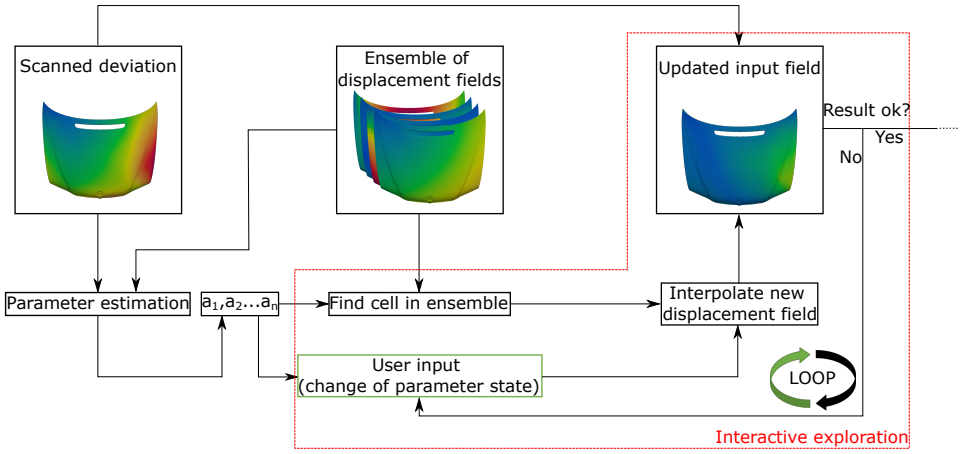


Figure 2 Detailed description of the workflow with attention to the interactive exploration loop marked in red. Inside this loop the user can change input parameters and gets an updated displacement field as response of the method.

3.1 Preparation steps

To be able to interact in real-time with measurement data, one-time preparation steps are necessary which are presented in this section. These preparations are already mentioned in Figure 1. First, an FE-simulation model must be created from a CAD-geometry for the part of interest. Also, the adjustable parameters and corresponding ranges must be defined and assigned to boundaries in the simulation model. Next, an ensemble of simulation runs needs to be created that samples the parameter space equally distributed. This ensemble can then be used to set up an estimator for solving the parameter estimation problem for a given deviation field. Different approaches can be used as kernel for the estimator. These already are highlighted in section 2. For this paper, the parameter estimation is considered as a problem that can be solved satisfactorily with methods already provided by the literature. With the estimator set up, the preparation steps are complete.

3.2 Mathematical Aspects

In this section the mathematical details of the interactive exploration method are explained, especially the data acquisition and the interpolation/approximation scheme are detailed in the following.

3.2.1 Data Acquisition

The generation of the pre-calculated simulation runs, especially the choice of how to sample the parameter space to achieve good interpolation results, is a crucial point. This data generation will be discussed for a three-dimensional case for demonstration purposes and is then extended to a six-dimensional parameter space to match up with the validations shown later in this paper.

Before we can perform interpolations on pre-simulated displacement fields, we need to perform multiple simulation runs to generate the necessary information. The amount of simulation runs $s \in \mathbb{N}$ that need to be performed, highly depends on the number of simulation parameters that are varied and on the sampling resolution $r \in \mathbb{N}$ of the parameter space

$X \subseteq \mathbb{R}^n$. For example: Let $\vec{a} \in X$ be a vector of one simulation parameter, where every parameter a_i can be varied in a range $[min_i, max_i]$ on its parameter space X_i.

$$\vec{a} = (a_1, a_2, ..., a_n), a_i \in [min_i, max_i]. \tag{1}$$

Then, the number of possible combinations of parameter states s for all axis of X can be calculated with:

$$s = r^n \tag{2}$$

Sampling Distribution

The sampling distribution of simulation runs presumed in equation 2 is equally spaced on every parameter axis $\{X_0, \ldots, X_n\}$ with the same amount of sampling points r per axis. The reason for choosing such a structured data set will be clarified in section 3.2.2. To get a better understanding of the structure of the resulting data set, Figure 3 shows an example with three simulation parameters and three sampling points per axis. The number of simulation

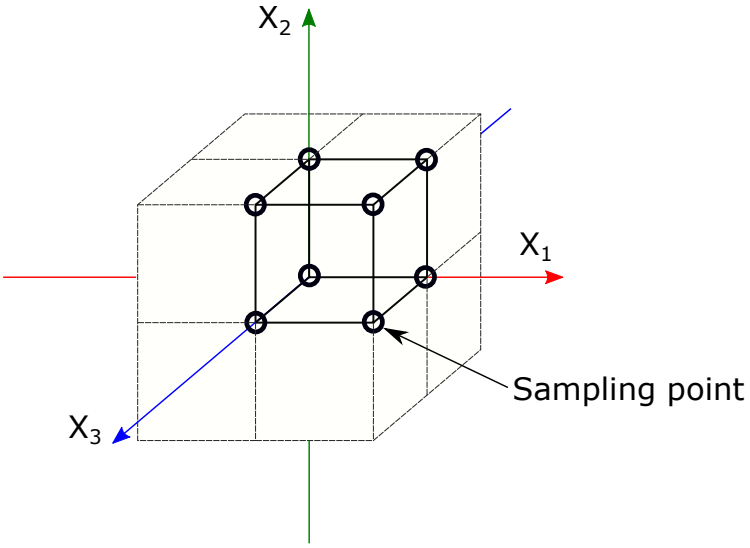


Figure 3 Simplified illustration in 3D of sampling one hypercube in a multidimensional parameter space.

parameters defines the order of dimension of the parameter space and the number of sampling points defines the number of (hyper)-cubes needed for filling the parameter space. In this particular example, we see three dimensions and three sampling points per axis resulting in eight cubes with a total of $3^3 = 27$ sampling points. One of these cubes and its sampling points (parameter 1,2,3 >= 0) is highlighted in the figure. Every sampling point in the parameter space is a combination of simulation parameters where an actual simulation run is performed, resulting in 27 simulation results which are the supports for the interpolation performed later on. As the simulation model of the use case presented in this paper has more than three adjustable simulation parameters, we next discuss how to apply this systematic sampling scheme on the general higher-dimensional case.

In the general case a geometric structure that subdivides a parameter space with the dimension n in an equally spaced manner, is called n-cube also named hypercube. In Table 1 the elements of the hypercubes up to $n = 6$ are listed. This table can be calculated by the

Table 1 Elements of an n-dimensional hypercube.

n	Name	0-face (vertex)	1-face	2-face	3-face	4-face	5-face	6-face	...
0	Point	1							
1	Line	2	1						
2	Square	4	4	1					
3	Cube	8	12	6	1				
4	Tesseract	16	32	24	8	1			
5	Penteract	32	80	80	40	10	1		
6	Hexeract	64	192	240	160	60	12	1	
⋮									

sequence A038207 that can be found in "The On-Line Encyclopedia of Integer Sequences®" (OEIS®) [1]. When we now consider six simulation parameters that are sampled with three points per axis we already need $3^6 = 729$ simulation runs that span 64 6D-hypercubes. Note that this exponential behavior for this systematic kind of sampling can result in a large number of simulation runs and thus in very expensive and time-consuming computations. To avoid an impossible amount of simulation runs, adaptive sampling strategies like provided by [23], can be applied. For the presented use-case no adaptive sampling was necessary.

3.2.2 Interpolation / Approximation Scheme

For generating information between sampled data points, we use interpolation. The interpolation scheme chosen in this paper for interpolating in a higher-dimensional space is called "inverse distance weighting". This scheme will now be explained in detail:

Let $\vec{a} = (a_1, a_2, ..., a_n)$ be a vector of parameters of one simulation and $\vec{u}(\vec{a})$ the corresponding simulation result represented by a displacement field, then we can express the pre-simulated displacement fields S as a set of simulation runs based on the parameter sets A:

$$A = \{\vec{a}_1, \vec{a}_2, \ldots, \vec{a}_m\} \tag{3}$$

$$S(A) = \{\vec{u}(\vec{a}_1), \vec{u}(\vec{a}_2), \ldots, \vec{u}(\vec{a}_m)\} = \{\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_m\} \tag{4}$$

Every parameter set contained in the quantity from equation 3 is considered as a known or sampled point in the parameter space. The corresponding displacement field can be interpreted as the "value" of this point.

Let $\vec{b} \in X$ be also a set of simulation parameters but $\vec{u}(\vec{b}) = \vec{u}_b \notin S(A)$, thus we consider $\vec{u}_b$ as an unknown value in the solution space $S(A)$ that needs to be interpolated. To do so, every known value is weighted by the inverse of the distance using the function $\lambda : X \times X \to \mathbb{R}$, calculated in the parameter space:

$$\vec{u}_b \approx \frac{1}{\sum_{l=1}^{m} \lambda(\vec{b}, \vec{a}_l)} \sum_{i=1}^{m} \lambda(\vec{b}, \vec{a}_i)\vec{u}(\vec{a}_i) \tag{5}$$

$$\lambda(\vec{b}, \vec{a}_i) = \frac{1}{D(\vec{b}, \vec{a}_i)} \tag{6}$$

$$D(\vec{b}, \vec{a}_i) = \|\vec{b} - \vec{a}_i\|_2 \tag{7}$$

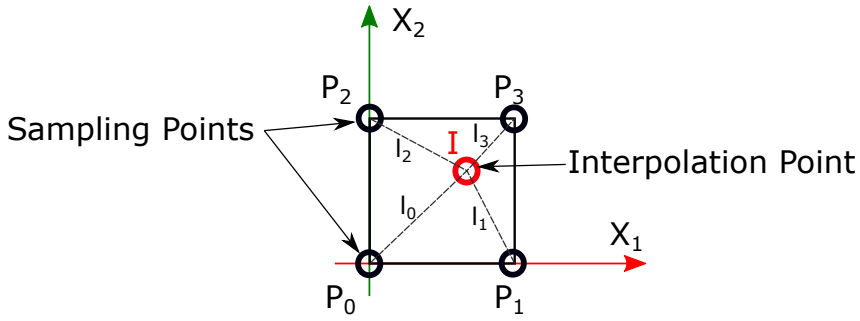


Figure 4 Calculation of the distances in the parameter space for an interpolation point (red) inside a sampled cell.

Equation 6 is a weighting factor by inverting the distance calculated between the point $\vec{b}$ and the sampled point $\vec{a}_i$, while

$$\frac{1}{\sum_{l=1}^{m} \lambda(\vec{b}, \vec{a}_l)} \tag{8}$$

is a factor to scale the sum of weights to 1 so that we can introduce the scaled weight $\lambda^*(\vec{b}, \vec{a}_i)$:

$$\lambda^*(\vec{b}, \vec{a}_i) = \frac{\lambda(\vec{b}, \vec{a}_i)}{\sum_{l=1}^{m} \lambda(\vec{b}, \vec{a}_l)} \tag{9}$$

To locally increase the weight of a sampled point, the inverse lengths are raised to the power of k:

$$\lambda^{*k}(\vec{b}, \vec{a}_i) = \frac{(\lambda(\vec{b}, \vec{a}_i))^k}{\sum_{l=1}^{m} (\lambda(\vec{b}, \vec{a}_l))^k}, where \sum_{i=1}^{m} \lambda^{*k}(\vec{b}, \vec{a}_i) = 1 \tag{10}$$

With the introduced scaled weights and the notation $\lambda_i^{*k} = \lambda^{*k}(\vec{b}, \vec{a}_i)$ we can express the interpolation as:

$$\vec{u}_b \approx \sum_{i=1}^{m} \lambda_i^{*k}(\vec{b}, \vec{a}_i)\vec{u}(\vec{a}_i) = \lambda_1^{*k}\vec{u}(\vec{a}_1) + \lambda_2^{*k}\vec{u}(\vec{a}_2) + ... + \lambda_m^{*k}\vec{u}(\vec{a}_m) \tag{11}$$

Figrue 4 demonstrates a simple 2D example the interpolation scheme. The red point I can be approximated by calculating the distances l_{0-3} and applying equation 11.

Weak Spots of Inverse Distance Weighting

When using inverse distance weighting, some aspects that are affecting the interpolation results are discussed in the following.

First, the distribution of the sampled points in the parameter space plays a significant role in this interpolation scheme. In Figure 5 an example is given for an uneven distribution of sampling points. The red point again is the point that needs to be interpolated. When performing inverse distance weighting on this kind of data set the interpolation result will be shifted towards the dense cluster of sampling points. Thus, the result of the interpolation would not be plausible anymore. Another factor that influences the interpolation result is the threshold radius when filtering the distances so that the interpolation does not have

to calculate weights and the linear combination of Equation 11 for the whole data set. Depending on the location of the interpolation point a slight change of the filter radius can result again in an even or uneven set of sampling points that are used for the interpolation. To prevent these shortcomings, the data set of pre-simulated displacement fields is distributed in an even manner like presented in Section 3.2.1. To prevent uneven distribution due to distance filtering, our method uses just the nodes of the hypercube where the interpolation node is located in. These sampling points are found by sorting the whole data set by distance and use the n closest points, where n is equal to the number of 0-faces of the corresponding hypercube, see Table 1.

Although there are more general approaches available in the branch of multivariate interpolation commonly used like kriging, spline interpolation, or radial basis function interpolation, inverse distance weighting is preferred in this paper. Different methods for scattered data interpolation are discussed and tested in [10]. The reason for choosing inverse distance weighting for solving the interpolation problem in the higher dimensional parameter space is the simplicity, robustness, and easy to implement nature of the algorithm. Also, the mentioned shortcomings of this algorithm can be prevented as discussed.

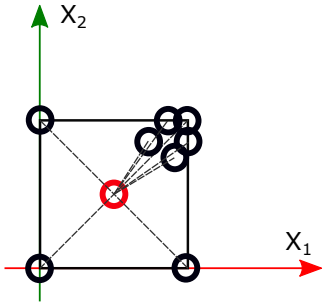


Figure 5 Influence of the distribution of sampling points. Local clusters lead to a higher weighting towards the cluster position.

3.3 Algorithm

In Algorithm 1, a pseudocode description of the interactive update is given. The input for this algorithm is an initial parameter set that is obtained from solving the parameter estimation problem for a measured deviation field. This initial parameter set is necessary to locate the measured deviations in the parameter space. Next, the measured displacement field is needed that will be updated by the algorithm. The third input is a list of parameters that were used to generate the pre-simulated displacement fields. To this list, the actual displacement fields are linked by a run-ID. The output of the algorithm is the updated measured displacement field.

At the beginning of the algorithm, a baseline is calculated by interpolating a displacement field for the initial parameter set. As we are interested in changes relative to this interpolation result, we need to subtract it from further interpolation results. For the interpolation, the steps that are described by the function “interpolate” have to be performed. First, the distances between the target point (in this case the initial parameter set) and every sampling point is calculated in the parameter space. Then the sampling points are sorted by distance and the 65 (64 points for the nodes of the 6D-hypercube and one point for the center of mass) closest points are used to perform the interpolation. For these 65 points, weights are calculated as described in Equation 11. Last, the weights are multiplied with the corresponding pre-

Algorithm 1 Pseudocode of Algorithm, structured by introducing two functions "calculate_weight" and "interpolate" that are used in the main body.

Data:
- $\vec{a}\_init$: Initial parameter set obtained from parameter estimator
- *measurement*: Measured displacement field
- *parameter_list*: Ensemble of pre-calculated displacement fields and corresponding list of simulation parameter

Result: Updated measured displacement field
Function `calculate_weight(`*distances*`):`
  # calculates the weight using Equation 11.
Function `interpolate(`*point*`):`
  distance_list=parameter_list - point
  distance_list.sort_by_distance()
  parameter_list.sort_by_distance()
  cut_off_list=distance_list[:65] # get the first 65 entries
  for *i in length(cut_off_list)* **do**
    weight=calculate_weight(cut_off_list[i])
    Output+=weight * simulation_result(parameter_list[i])
  end
  return Output
Main
baseline=interpolate($\vec{a}_{init}$)
#interactive Loop
while *TRUE* **do**
  get user_input
  interpolation_result=interpolate(user_input)
  updated_measurement=measurement - baseline + interpolation_result
  render(updated_measurement)
end

simulated displacement fields. The weighted displacement fields are summed up and returned by the function. After the baseline is calculated, the interactive loop starts, where the user can make changes in the form of a parameter set. This input parameter set is then also interpolated by the same function "interpolate". The measured displacements are then updated by subtracting the baseline and adding the interpolation result of the user input resulting in an approximated displacement field for the new parameter state. Finally, the updated field is rendered. The interactive loop runs until it is stopped by the user.

4 Use Case

We present a real-world use case scenario with validations performed on simulation and experimental data sets.

4.1 Simulation Setup

The chosen part is the engine hood of a car body. This example represents an assembly consisting of seven individual parts. The assembly structure is shown in Figure 6. Besides the visible outer skin, two inner parts are visible when lifting the hood. Four thicker sheet

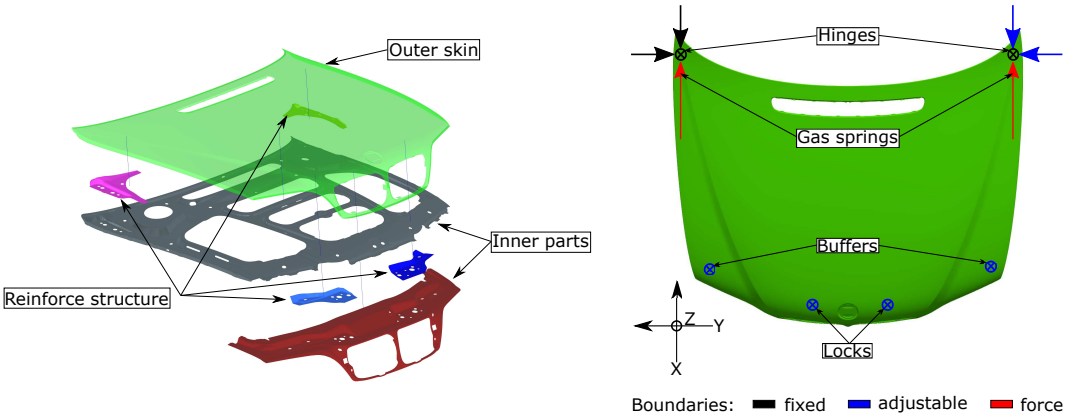


Figure 6 Left side: assembly structure of used hood. Right side: boundaries used for simulation with the coordinate system. The boundaries are "fixed" (cannot move at all), "adjustable" (to manipulate hood shape), or "force" (by using defined forces to model gas springs attached to the hood).

Table 2 Simulation parameters and corresponding value ranges.

Simulation parameter	Label	Description	Value range [mm]
a_1	HINGE_X	Hinge left, X-direction	[-1;1]
a_2	HINGE_Y	Hinge left, Y-direction	[-1;1]
a_3	BUFFER_L	Buffer left, Z-direction	[-1;1]
a_4	BUFFER_R	Buffer right, Z-direction	[-1;1]
a_5	LOCK_L	Lock left, Z-direction	[-1;1]
a_6	LOCK_R	Lock right, Z-direction	[-1;1]

metal parts are inside the hood to reinforce its structure. An FE simulation model based on the CAD geometry is created. Every component of the hood has meshed with first-order 3D thin-shell elements. The total number of mesh nodes is 365,683 and the number of mesh elements is 365,154. All components are joined together by modeling all spot welds as well as structural and acoustic adhesives. Finally, mechanical boundaries are added, as shown in Figure 6. This figure introduces the chosen coordinate system (X-, Y-, and Z-directions). The hood is attached with two hinges to the chassis. The right hinge (direction-of-travel) is held in position while the left hinge is adjustable in X- and Y-directions. In the closed state one gas spring per side is pushing with 580N in X-Z-direction against the hood near the hinges. In the front, two locks and two buffers are modeled as boundaries in Z-direction. While the buffers can only push against the hood, the locks can push and pull in both Z-directions. All boundaries are applied to the inner parts or the reinforced structure. Figure 6 shows an overview of the locations of all boundaries and Table 2 lists the adjustable boundaries with the corresponding value ranges. Based on the adjustable boundaries and the range of values, the data set of pre-simulated displacement fields is generated. We sample three points per axis. The number of simulation parameters is six, and by applying Equation 2 a total of 729 simulation runs are performed, used to define 64 6D-hypercubes. Further, 64 additional simulation runs are performed by sampling the center point of every 6D-hypercube. This approach produces a denser sampling without affecting the underlying data structure and not affecting interpolation result negatively.

4.2 Validation with Simulation Data

To assess the performance of the interpolation using inverse distance weighting, a validation with simulation data is performed. The goal of this validation is to understand the scale of the interpolation error. A "worst-case scenario" is set up by measuring the interpolation quality of a point in parameter space that is as far away from all sample points as possible. Therefore, we exclude the 64 simulation runs from the data set that were generated at the center point of the 6D-hypercubes. Instead, we interpolate the data values for these center points to compare the result with the excluded simulation-based value. The result is shown in Figure 7 (parameter colored in red) for one hypercube. We argue that for every other interpolation point inside a 6D-hypercube, the result generated via inverse distance weight interpolation would be better, as the interpolation location will be closer to one of the 6D-hypercube's sampled corner points. Additionally, the samples of the center points are part of the data set for further interpolations. To show that the quality of interpolation improves for different parameters, random parameter sets are generated in the allowable range specified in Table 2 and full simulation runs are performed. An interpolation result is calculated for the same parameter sets. The simulation results are compared with the interpolation results. This comparison is shown in Figure 7 for three randomly generated parameter sets. The first row in the figure shows the simulation parameters used for simulating/interpolating the results for each column. Rows two and three show these results (displacement magnitude) as color-coded images. As differences are difficult to see when visually comparing these two rows, the last row shows the error distribution of the comparison of the simulated-based and interpolated displacement fields. For all three examples, the RMS error is below the worst-case scenario.

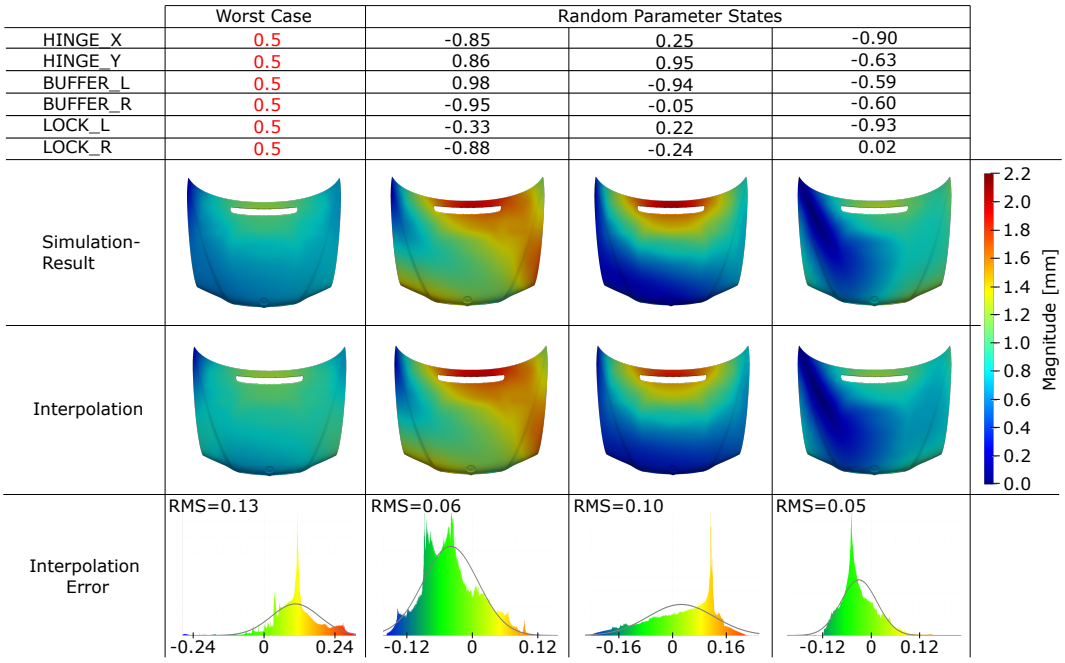


Figure 7 Comparison of simulation and interpolation. Top row: used parameter (red = worst-case scenario) to generate the simulation-based and interpolation results; second/third row: calculated displacement fields; bottom row: histogram for distribution of interpolation error evaluated at all vertices, RMS[2] error value.

[1] Root-Mean-Square

4.3 Validation with Experimental Data

The quality of the interpolation result is crucial for practical use. The results presented in Section 4.2 show promising performance concerning the precision, but further investigation with experimental data should be made. We describe our experimental setup and present results.

4.3.1 Hardware Set-up

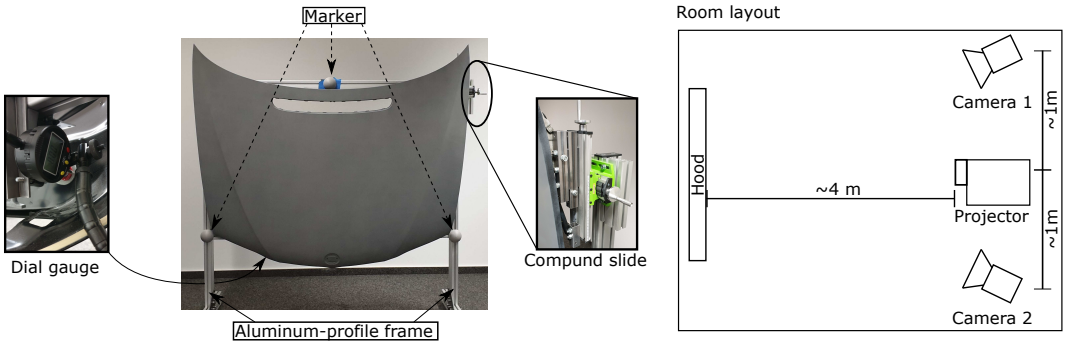


Figure 8 Left-hand side: experimental setup used for validation against measurement data; right-hand side: set-up used for scanning the engine hood.

The hood is prepared for measurement experiments by applying a light gray paint to it. This is necessary to generate a 3D scan with a structured light scanner. Besides the part itself, a fixture is designed and built from aluminum profiles. On top of this aluminum frame the locks, buffers, hinges, and gas springs are mounted. The set-up is depicted in Figure 8. To make the relative positions of the hinges adjustable, the left hinge is mounted onto a compound slide. The locks and buffers have proprietary adjustment options. The frame is equipped with markers to align different measurements. A dial gauge is used to change boundaries precisely. The hood is not mounted in the orientation as it would be in the assembled in a car, but this fact only affects the direction of gravity considered in the simulation. For the purpose of our validation, the orientation of the hood does not matter. Nevertheless, we point out that orientation matters when considering the scanning set-up, as depicted in Figure 8 on the right-hand side. The cameras and the projector of the structured light scan system must have a certain distance to the scanned object to capture it in its entirety. Usually, complex parts are captured with many pictures that are later stitched together. However, to reduce uncertainty caused by data registration, we capture every measured state of the hood in a single picture. This approach results in a relatively coarser resolution and noisier measurement data, but the quality of the result data is still sufficiently precise for performing our validation. A significant advantage of this set-up is the fact that the hood and the hardware components of the 3D scanner are not moved between scans. All measurements are made in the same coordinate system. This method almost removes the need for alignment. The markers mounted to the fixture are primarily used to determine whether something has moved between scans. (The measurement system used is a modified HP pro S3 structured light scanner, calibrated with custom calibration panels for the large size of the scan window.)

4.4 Experimental Validation Results

To assess the performance of the entire workflow, presented in Figure 2, a prototype implementation of Algorithm 1 is discussed next. We consider optical measurement data obtained for different boundary situations. Figure 9 shows the sequence of steps of the validation. First, a reference measurement is taken, and geometrical changes are computed relative to

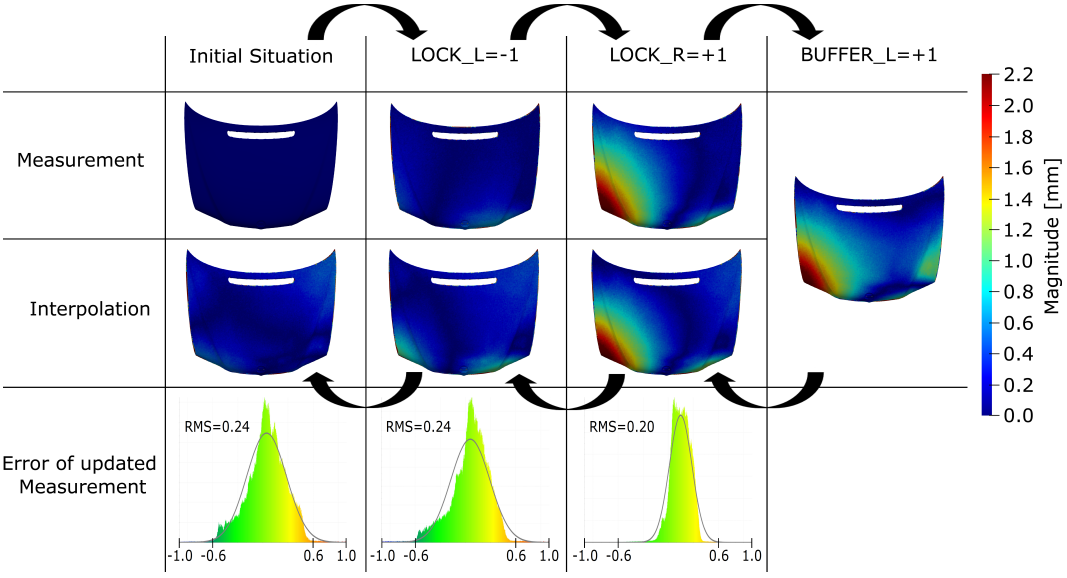


Figure 9 Experimental results. Top row: measured displacements caused by adjusting boundaries, compared to reference measurement (dark blue, leftmost); second row: interpolation results reconstructed "backward" from the measured displacement field (rightmost); third row: histogram for distribution of error caused by measurement update, evaluated at all vertices, and RMS error value.

this reference model (dark blue - top left). Single boundaries are changed, one after the other (top row). After every change of boundaries, the new geometry is captured by a scan. The resulting geometries are compared relative to the reference measurement, producing the displacement fields shown in the first row of the figure. The displacement field in the rightmost column is used as an input field for interactive interpolation. To validate the proposed method, Algorithm 1 is used to calculate the measured states, based on the displacement field. To reduce the influence of an interpolation error, only boundary states that match a sampled point in the pre-calculated simulation ensemble are used. When traversing the second row from right to left, we see the results produced by the proposed algorithm. The initial displacement field is updated by the change of the interpolation towards the geometry of the reference measurement, where the displacement should ideally be zero for every vertex. It can be observed that the resulting displacement fields do not match the measured data exactly.

4.5 Run Times

To assess the interactive aspect of our method, we measured run times, listed in Table 3. The results show that creating the simulation data ensemble – which is a one-time pre-computation step – requires about 103.5 hours. Nevertheless, as this is a pre-computation step that can

also be performed in parallel, it scales with the available computational resources. The last column, "Interpolation", is the relevant column, it documents interactive response capability of our presented method; the average computational time is only 0.2s.

Table 3 Computational times.

	Simulation Ensemble	Comparison of measurements	Interpolation
Computational time	793 x 470s = 103.5 h	2s	0.2s

5 Discussion

This paper addresses different aspects of realizing its goal. Thus, the following topics from the presented use case scenario are discussed in this section: The data generation of the pre-calculated simulation ensemble, the results from the validation with simulation data as well as the results from the validation with experimental data, and finally, the achieved run times.

To generate sampling data for a higher dimensional parameter space can result in a large number of necessary simulation runs. Already for the shown use case of six parameters, a coarse sampling resulted in 793 runs. As the range of values is small relative to the size of the part, a linear behavior can be presumed, making a coarse sampling sufficient. For the use case of tuning in gap and flushness in sheet metal assemblies, this assumption is appropriate. For parts that behave nonlinearly like ones that are made of composite materials, different sampling and interpolation strategies might be necessary.

The interpolation/approximation scheme used in this paper, to achieve the results presented in Section 4.2, is very simple and easy to implement. Although the achieved results are sufficient for the presented use case it is hard to generalize this specific approach to other use cases. The results in Figure 7 show an interpolation error (RMS) up to 0.1 (worst case excluded). Considering the used sampling grid resolution of $1mm$ on each parameter axis, extended by the additional sampling at the center of mass for each hypercube, this error is indeed not negligible. Nevertheless, for exploitative purposes we consider the achieved precision as sufficient, as the goal of the interpolation is not to replace simulation runs, but to provide an impression of what's happening in between the sampled points. When the user visually compares the simulation and interpolation result, it is hard to spot out differences, so we can argue that the achieved precision is sufficient for the presented use-case.

When using experimental data, things become more complicated. The results presented in Figure 9 show on the same scale visible differences. To justify these higher errors, different sources of errors were identified. First, every measurement system has uncertainties. The measurement system includes the 3D-scanner itself, the fixture, and environmental influences. The 3D-scanner is capable of providing scans with a precision of up to $0.05mm$. But as the measurement conditions were not ideal (compare Section 4.3.1) we assume a higher uncertainty, estimated at $0.1mm$. The second source of error is the fixture which is assumed as entirely rigid. During the experiments, it turned out that some boundaries show a slight runaway when changing the boundary state. This was recognized and re-adjusted, but anyway causes errors when comparing set parameters. Besides uncertainties caused by the physical setup, there are also sources of errors in the workflow of processing the measurement data. For example, smoothing and outlier removal as well as solving the registration problem to match measurement and simulation vertices is a crucial step. For this experiment, the assumption was made, that only displacements perpendicular to the surface of the part occur

as in-plane movements cannot be captured by a 3D-scan. The registration was simply made by finding the closest point when comparing two meshes. This assumption might lead locally to large errors. Nevertheless, when visually comparing the real measurements and due to interpolation updated measurement, we see highly similar behavior. So we can argue that the goal of providing information about the behavior of the measured part due to boundary changes is achieved.

Last, the interactive aspect needs to be addressed. Therefore, the run times of different tasks were recorded and listed in Table 3. The crucial step is the calculation of the interpolation result, which takes on average $0.2s$. This value meets the pre-defined design goal of below $0.5s$.

To summarize the discussion, we can say that the proposed method works satisfyingly, although individual components could be improved. Especially for the interpolation scheme, as well as for the point cloud registration problem, other methods might improve the overall result.

6 Conclusions

We have introduced an interactive approach for exploring deviation measurements of sheet metals, to help with the determination of viable boundary adjustments necessary to eliminate or significantly reduce deviations. Our improvement of existing methods is the design and implementation of an interactive system framework that can be utilized intuitively by an inspection engineer to identify quality shortcomings and quickly enforce countermeasures to positively affect sheet metal behavior in an assembly. Our system was driven by the design and quality objectives employed in today's automotive industry. Our system is especially helpful to an inspection engineer during the early stages of production when it is desirable to evaluate multiple combinations of boundary adjustments in a small amount of time. As our interactive system supports the real-time exploration of such combinations, it represents a substantial acceleration of the decision-making process thereby reducing time and cost. We have validated our methods and prototype system by applying them to simulated and experimental data. Our validation results support the effectiveness of our system for real-world scenarios.

Regarding potential directions for future research, it is possible to consider the use of other data interpolation or approximation schemes, instead of inverse distance weighting. Further, other point cloud registration methods could be utilized.

References

1 OEIS Foundation Inc. (2020). *The On-Line Encyclopedia of Integer Sequences*, 2020. URL: https://oeis.org/A038207.

2 Manoj Babu, Pasquale Franciosa, and Dariusz Ceglarek. Spatio-temporal adaptive sampling for effective coverage measurement planning during quality inspection of free form surfaces using robotic 3d optical scanner. *Journal of Manufacturing Systems*, 53:93–108, 2019. doi:10.1016/j.jmsy.2019.08.003.

3 Erin Santini Bell, Masoud Sanayei, Chitra N. Javdekar, and Eugene Slavsky. Multiresponse parameter estimation for finite-element model updating using nondestructive test data. *Journal of Structural Engineering*, 133(8):1067–1079, 2007. doi:10.1061/(ASCE)0733-9445(2007)133:8(1067).

4 T. Bui-Thanh, M. Damodaran, and K. Willcox. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA Journal*, 42(8):1505–1516, 2004. doi:10.2514/1.2159.

5 Jaime Camelio, S. Jack Hu, and Dariusz Ceglarek. Modeling Variation Propagation of Multi-Station Assembly Systems With Compliant Parts . *Journal of Mechanical Design*, 125(4):673–681, January 2004. doi:10.1115/1.1631574.

6 Jaime A. Camelio, S. Jack Hu, and Dariusz Ceglarek. Impact of fixture design on sheet metal assembly variation. *Journal of Manufacturing Systems*, 23(3):182–193, 2004. doi:10.1016/S0278-6125(05)00006-3.

7 Johan S. Carlson and Rikard Söderberg. Assembly root cause analysis: A way to reduce dimensional variation in assembled products. *International Journal of Flexible Manufacturing Systems*, 15(2):113–150, 2003. doi:10.1023/A:1024453207632.

8 F Claus, H Hagen, H Leitte, and B Hamann. Decomposing deviations of scanned surfaces of sheet metal assemblies[submitted]. *Journal of Manufacturing Systems*, 2020.

9 Christopher Fleischmann, Irina Leher, Reinhold Hartwich, Marc Hainke, and Stefan Sesselmann. A new approach to quickly edit geometries and estimate stresses and displacements of implants in real-time. *Current Directions in Biomedical Engineering*, 5(1):553–556, 2019. doi:10.1515/cdbme-2019-0139.

10 Richard Franke. Scattered data interpolation: Tests of some method. *Mathematics of Computation*, 38(157):181–200, 1982. URL: http://www.jstor.org/stable/2007474.

11 A. S. Gendy and A. F. Saleeb. Nonlinear material parameter estimation for characterizing hyper elastic large strain models. *Computational Mechanics*, 25(1):66–77, 2000. doi:10.1007/s004660050016.

12 Stratos Idreos, Olga Papaemmanouil, and Surajit Chaudhuri. Overview of data exploration techniques. In Timos Sellis, Susan B. Davidson, and Zack Ives, editors, *Compilation proceedings of the 2015 ACM Symposium on Principles of Database Systems, ACM SIGMOD International Conference on Management of Data, and SIGMOD/PODS 2015 PhD symposium, May 31 - June 4, 2015, Melbourne, VIC, Australia*, pages 277–281, New York, NY, 2015. ACM. doi:10.1145/2723372.2731084.

13 Michael L. Johnson and Lindsay M. Faunt. Parameter estimation by least-squares methods. In *Numerical Computer Methods*, volume 210 of *Methods in Enzymology*, pages 1–37. Academic Press, 1992. doi:10.1016/0076-6879(92)10003-V.

14 Gergely Kristóf and Bálint Papp. Application of gpu-based large eddy simulation in urban dispersion studies. *Atmosphere*, 9(11):442, 2018. doi:10.3390/atmos9110442.

15 Xiaoyun Liao and G. Gary Wang. Simultaneous optimization of fixture and joint positions for non-rigid sheet metal assembly. *The International Journal of Advanced Manufacturing Technology*, 36(3):386–394, 2008. doi:10.1007/s00170-006-0827-5.

16 S. C. Liu, S. J. Hu, and T. C. Woo. Tolerance Analysis for Sheet Metal Assemblies. *Journal of Mechanical Design*, 118(1):62–67, March 1996. doi:10.1115/1.2826857.

17 Cong Lu and Hong-Wang Zhao. Fixture layout optimization for deformable sheet metal workpiece. *The International Journal of Advanced Manufacturing Technology*, 78(1):85–98, 2015. doi:10.1007/s00170-014-6647-0.

18 Kuan Lu, Yulin Jin, Yushu Chen, Yongfeng Yang, Lei Hou, Zhiyong Zhang, Zhonggang Li, and Chao Fu. Review for order reduction based on proper orthogonal decomposition and outlooks of applications in mechanical systems. *Mechanical Systems and Signal Processing*, 123:264–297, 2019. doi:10.1016/j.ymssp.2019.01.018.

19 Yuki Mori and Takeo Igarashi. Plushie. *ACM Transactions on Graphics*, 26(99):45, 2007. doi:10.1145/1239451.1239496.

20 Giulio Reina, Matilde Paiano, and Jose-Luis Blanco-Claraco. Vehicle parameter estimation using a model-based estimator. *Mechanical Systems and Signal Processing*, 87:227–241, 2017. Signal Processing and Control challenges for Smart Vehicles. doi:10.1016/j.ymssp.2016.06.038.

21 Evan Strasnick, Maneesh Agrawala, and Sean Follmer. Scanalog. In Krzysztof Gajos, Jennifer Mankoff, and Chris Harrison, editors, *Proceedings of the 30th Annual ACM Symposium on*

User Interface Software and Technology, pages 321–330, New York, NY, USA, 10202017. ACM. doi:10.1145/3126594.3126618.

22 Mario Teixeira Parente, Steven Mattis, Shubhangi Gupta, Christian Deusner, and Barbara Wohlmuth. Efficient parameter estimation for a methane hydrate model with active subspaces. *Computational Geosciences*, 23(2):355–372, 2019. doi:10.1007/s10596-018-9769-x.

23 Steven K. Thompson. Adaptive cluster sampling. *Journal of the American Statistical Association*, 85(412):1050–1059, 1990. doi:10.1080/01621459.1990.10474975.

24 J. Whyte, N. Bouchlaghem, A. Thorpe, and R. McCaffer. From cad to virtual reality: modelling approaches, data exchange and interactive 3d building design tools. *Automation in Construction*, 10(1):43–55, 2000. doi:10.1016/S0926-5805(99)00012-6.

25 F. Yoshida, M. Urabe, and V.V. Toropov. Identification of material parameters in constitutive model for sheet metals from cyclic bending tests. *International Journal of Mechanical Sciences*, 40(2):237–249, 1998. doi:10.1016/S0020-7403(97)00052-0.

26 A. Zerwer, G. Cascante, and J. Hutchinson. Parameter estimation in finite element simulations of rayleigh waves. *Journal of Geotechnical and Geoenvironmental Engineering*, 128(3):250–261, 2002. doi:10.1061/(ASCE)1090-0241(2002)128:3(250).

Influence of Flank Face Structuring on Cooling, Tool Lifetime and Borehole Quality When Drilling Inconel 718: Physical Simulations and Experimental Validation

Daniel Müller
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Benjamin Kirsch
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Jan C. Aurich
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Abstract

When drilling difficult-to-cut materials such as Inconel 718, the drills are exposed to high thermo-mechanical loads. Due to the low thermal conductivity of the workpiece material, a large amount of the generated heat has to be dissipated by the metal working fluid (MWF). However, the cutting zone is located inside the workpiece, which makes it challenging to provide sufficient MWF to the cutting zone. To solve this, drills with internal cooling channels are commonly used. In this work, the influence of differently structured flank faces on cooling efficiency, tool life, process forces and borehole quality is investigated. The influence of the structures on the cooling was investigated by Computational-Fluid-Dynamics (CFD) simulations. These simulations allow a detailed analysis of the flow conditions inside the borehole and showed that the structuring improved flow conditions, especially near the thermally highly loaded main cutting edge. The improved flow conditions resulted in an extension of the tool life by up to 22 % compared to unstructured drills in experimental investigations.

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases drilling, cooling, CFD

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.7

Funding The investigations presented here partly result from the research project IGF 19154N of the German Machine Tool Builders' Association (VDW). It was funded within the program for industrial collective research (IGF) by the Federal Ministry of Economics and Technology (BMWi) via the German Federation of Industrial Research Associations (AiF). This research was also partly funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057.

1 Introduction

The nickel-based alloy Inconel 718 is one of the most widely used superalloys. It was developed for applications in aerospace turbines, but is also used in the automotive and chemical industries, due to its excellent material properties [4]. Inconel 718 provides superior mechanical properties such as high tensile, fatigue and fracture strength even at temperatures above 700°C. During machining, these material properties as well as the tendency to work hardening lead to high thermal and mechanical loads on the tools and consequently to a low productivity and a limited tool life [4, 21].

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 7; pp. 7:1–7:17

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Drilling is one of the oldest and most important machining processes. With regard to the process conditions, the process has some specific characteristics: an unfavorable heat distribution in the cutting zone, a cutting speed that drops to zero towards the axis of rotation, the rubbing of the chips as well as the rubbing of the margin against the surface of the borehole as well as a difficult evacuation of the chips. Due to the unfavorable heat distribution in the cutting zone and the nearly complete conversion of the mechanical energy applied for cutting into heat, the tools are exposed to high thermal loads [10]. As the strength of carbides decreases with increasing temperature, the wear resistance also decreases with increasing temperature [10]. To counteract this, metalworking fluids (MWF) are used to reduce the thermal load and consequently the wear of the drill. Besides the increase of the tool life, an efficient supply of the cooling lubricant to the cutting zone also leads to an improved machining quality [10, 20].

The removal of chips through the chip flutes of the drills and the position of the cutting zone inside the workpiece make an external coolant supply difficult. For this reason, in industry internal coolant supply is used, especially for deep bores (length/diameter > 3). The MWF is supplied via channels inside the tool. The outlet positions of the helical channels are on the flank faces of the drill and lead to an indirect cooling of the cutting zone. This results in improved cooling of the tool and the workpiece. Furthermore, the chip removal is supported and the friction between tool and chips is reduced [10]. The productivity of drills can be increased by a more efficient cooling lubrication by means of optimized internal cooling [5]. To improve internal cooling conditions, different design elements can be used. Previous investigations on internal MWF supply have focused on the flow behaviour within the cooling channel and the flute [9, 17]. Oezkaya et al. [15] analyzed the influence of the cooling channel diameter and the supply pressure on the cooling efficiency. Their findings showed that especially a higher supply pressure improves the cooling. Also, the integration of structures into the flank face near the main cutting edge can lead to improved cooling lubrication [3, 7, 16]. By adding grooves in the flank face, Guo et al. were able to minimize the feed forces and wear [7]. Beer et al. were able to achieve tool life increases up to 50 % by adding a groove-shaped structure in the first flank face. Furthermore, the structuring led to an improved borehole and surface quality [3].

In our previous works, the influence of the cooling channel outlet position at the flank face was investigated by numerical flow simulations and experiments. It was shown that cooling channel outlets near the main cutting edge lead to an improved cooling [5, 6]. The work in [14] and [13] focused on the influence of the second clearance angle, the number of cooling channels as well as the cooling channel diameter with respect to the thermal load of the tools. An enlargement of the cooling channel diameter and a decrease of the second clearance angle led to better cooling [13], whereas an increase in the number of cooling channels resulted in a deterioration of the cooling. Simulative investigations in [12] showed that the integration of structures in the first flank face can improve the flow conditions and thus also the efficiency of the cooling. Furthermore, it was shown that the influence of the structural shape on the cooling is more significant than the influence of the increased surface and the increased flow volume [12].

In this paper, an approach for the simulative investigation of internally cooled drills is presented and the influence of structured flank faces on the cooling of the drill, tool life, process forces and borehole quality have been investigated.

2 Experimental design

2.1 Machining Setup

Drilling was performed on a 5-axis machining centre (DMU 70 eVolution[1]) with the experimental setup shown in Figure 1a. Blind holes with a depth d_h of 24 mm were drilled at a cutting speed v_c of 20 m/min and a feed rate f of 0.09 mm/rev. A constant MWF volume flow rate $\dot{V}$ of 1 l/min (emulsion, 6 %) was chosen. The experiments were carried out on Inconel 718. The tools used were uncoated cemented carbide (K40UF: 10 % CO and 90 % WC) twist drills with a drill diameter d of 8 mm. All drills had a point angle σ of 140°, a twist angle ϵ of 30°, a first clearance angle α_1 of 10° and a second clearance angle α_2 of 25°. The distance from the center of the cooling channels to the center of the drill r_c was 3.8 mm, and the cooling channel diameter d_c was 1 mm. The variations of the structuring can be seen in Figure 1b. All structures had a spacing of 150 µm to the main cutting edge, a depth of 55 µm and were created by laser.

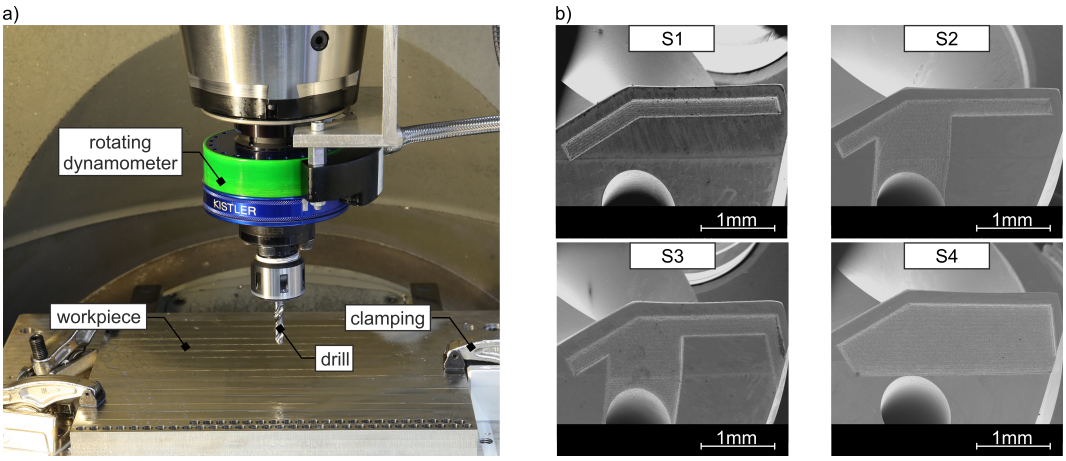


Figure 1 a) Experimental setup b) Structured drills.

The structures S1 to S3 are groove-shaped cavities along the cutting edge. The shape of the grooves was adapted to the shape of the main cutting edge to improve the flow conditions in this area. Based on S1, a connection between the cooling channel and the structure was implemented for S2 - S3. For S3 an additional slot was added near the cutting edge corner. The intention of the additional slots is to improve the fluid flow into the grooves. For structure S4, almost the entire area of the first flank face and parts of the second flank face were lowered.

2.2 Characterization methods

The forces in x-, y- and z-direction and the torque during drilling were measured using a rotating dynamometer (Kistler 9123 C[1]) with a sampling rate of 2 kHz. From each drill variant, two drills were used. To determine the forces, the torque and the borehole quality, measurements were taken at each data point over 3 consecutive boreholes per tool. Averaging was therefore carried out over 6 measurements.

The maximum width of flank wear land depending on the drilling path was determined through macroscopic images. The maximum width of flank wear land was evaluated at all four cutting edges per variation (2 tools for each variant) and the values were averaged. The

uncertainty of the measurement results is given by the average standard deviation $\overline{\sigma}$. To obtain the average standard deviation, the standard deviations were first calculated at every data point from the four measurements of each drill variant. Then, the arithmetic mean of all standard deviations for the respective variable was calculated.

The borehole quality was determined by evaluating the diameter, the concentricity and the roundness of the boreholes with a 3D coordinate measuring device (TESA Micro-Hite 3D dcc[1]). Nine measurement planes were placed in the boreholes every 2 mm. For each measurement plane, nine equally distributed measurement points were used. The first measurement plane of the respective borehole served as reference plane for the calculation of the concentricity. In order to consider the influence of the drill diameter on the diameter of the borehole, the borehole diameters were normalized according to:

$$d_{borehole} = \frac{\overline{d}_{borehole}}{d_{drill}} d_{drill}^{ideal} \tag{1}$$

where $\overline{d}_{borehole}$ is the measured borehole diameter, d_{drill} is the measured drill diameter and d_{drill}^{ideal} is the ideal drill diameter (8.0 mm). The surface roughness of the boreholes was determined by a stylus instrument (MarSurf XR20 GD120[1]) according to DIN EN ISO 4288 both at the entrance and the end of the borehole by means of three measurements over the circumference of the borehole. The surface roughness was specified via the average surface roughness R_z and the maximum surface roughness R_{max}.

The static pressure of the MWF within the collet before entering the drill was measured with a pressure sensor (Entran EPX-N01-70B[1]).

3 Numerical Simulation

Computational Fluid Dynamics (CFD) simulations have been proved to be a useful tool to analyze and visualize the flow conditions of internally cooled drills [12, 15]. The approach for such simulations is divided into the following five steps: Creation of the geometry (CAD model), discretization of the CAD model (meshing), definition of the boundary conditions, iterative computation and evaluation of the simulation results. The simulations presented here were carried out with Ansys CFX[1].

3.1 CFD fundamentals

The basic equations in numerical flow computations are the conservation laws of mass, momentum and energy. These equations describe all flows of Newtonian fluids. However, the computation workload, especially for turbulent flows, can become very high, due to the non-linearity. Therefore, most CFD programs are based on the Reynolds-averaged Navier-Stokes equations. These equations represent the physics with sufficient accuracy and acceptable computing time. Small turbulences are not resolved, but modelled by turbulence models. In addition to a suitable turbulence model, the boundary conditions at the inlet and outlet must also be selected carefully [8].

3.2 Drill geometry and fluid volume

The basis for the simulations were the CAD models of the examined drills. A scan of the reference tool was used to create a parameterizable CAD model of the drill geometry by reverse engineering. By scanning the tool, the exact geometry of the drill can be used within the simulations, which allows a comparison of the simulated findings with the experimental

results. To reduce the computational time, only the drill tip with a length of 10 mm (model reduction) was considered within the simulations. The structures were implemented in accordance with the real tools. Besides the drills, the respective complementary fluid volumes, which represent the flow area within the simulation, were also created by means of a Boolean operation.

3.3 Mesh

The quality of the mesh is fundamental for the accuracy, the reliability and computing time of the simulation [8]. A high-quality mesh implies refinements in areas of interest, a good aspect ratio, and only as few cells as necessary. Due to the complex geometry of the tool and the fluid volume, unstructured meshes of tetrahedron elements with a size of 2.0 µm up to 400 µm were used. The mesh of the fluid volume was refined (2-7 µm) in particular near the main cutting edge, at the cooling channels, at the structures and in the area of the heat input. Special attention was given to the area of the fluid volume near the main cutting edge between the flank face and the bottom of the bore hole. Due to the narrow space in this area, an inflation layer was applied in addition to the refinement (Figure 2).

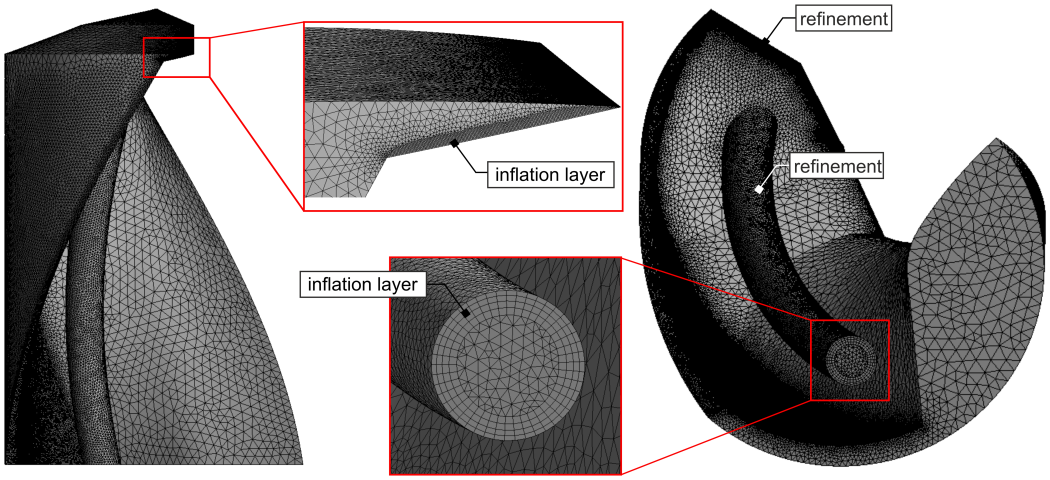


Figure 2 Mesh of the fluid volume.

3.4 Transferred boundary condition

Due to the model reduction described in Section 3.2, the results could differ from those of a complete model. However, the calculation of a complete model would be very time-consuming. To exclude possible discrepancies a second simulation model was used to generate a transferred boundary condition [2]. For this purpose, a model was built which calculates the flow conditions of the complete model, whereby e.g. the drill geometry and thermal effects are neglected for reasons of computing time. The flow conditions at different positions of the model can then be exported and used as a boundary conditions (transferred boundary condition) for all the simulations with the reduced models. As a result, all simulations can be carried out with a better accuracy and an overall reduced computing time. The model used maps the geometry of the cooling channels of the drill and the interior of the chuck up to the location where the fluid pressure is measured (Figure 3a). At the inlet of the model, the determined pressure (3.3 bar) was used as the boundary condition (total pressure) and at the

outlet the corresponding volume flow rate (1 l/min). An unstructured mesh of tetrahedron elements with a size of 5.0 µm up to 600 µm was used. The cooling channels were refined and an inflation layer was applied (Figure 3b). To model the turbulences, the κ-ω-SST (Shear Stress Model) was used [5, 15]. This turbulence model combines the advantages of the most frequently used turbulence models κ-ω and κ-ε. As a result, the turbulences near the wall as well as far from the wall are represented most realistically [19]. The convergence criterion of the simulation was defined to be 10^{-5}.

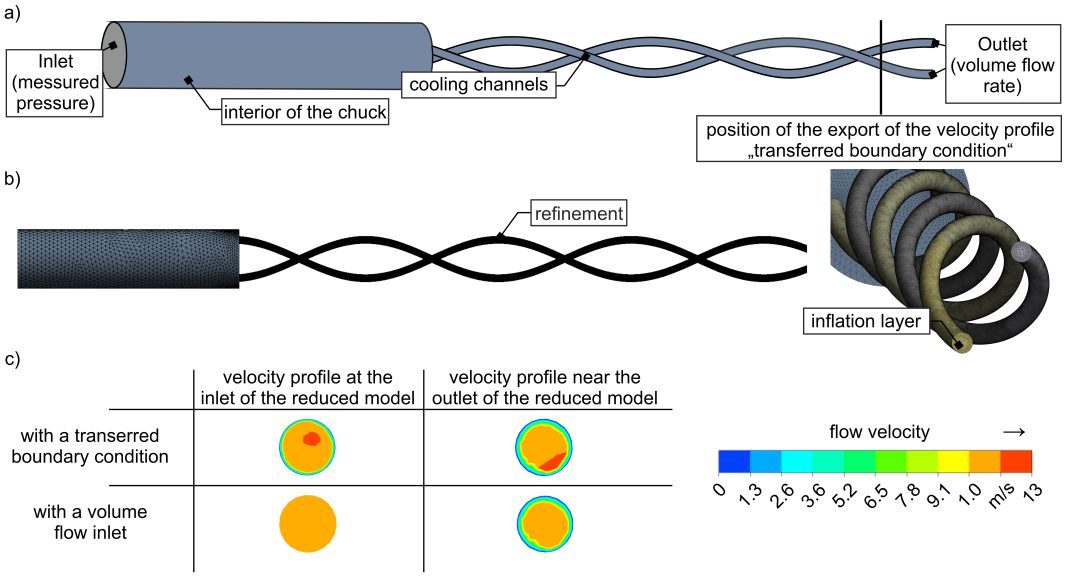


Figure 3 a) Model to simulate the flow conditions inside the cooling channels b) Mesh of the model c) Influence of the inlet boundary condition on the flow inside the cooling channel.

Figure 3c shows the flow profiles of the reduced model at the inlet of the cooling channel and near to the outlet of the cooling channel. The upper line contains the resulting flow profiles when using a transferred boundary condition, the lower line shows the profiles when using a volume flow inlet. Applying the transferred boundary condition, a completely defined flow profile is obtained at the inlet, whereas using a volume flow inlet results in a constant flow velocity over the inlet. Even near the cooling channel outlet, differences in the flow conditions are still apparent, which could influence the results regarding the cooling of the tool.

In order to transfer the resulting flow conditions of the complete model to the reduced model, the velocity profile was exported from the complete model at the position of the inlet of the reduced model and then used in the reduced model as the inlet boundary condition.

3.5 Boundary conditions

The κ-ω-SST (Shear Stress Model) was used to model the turbulence. The boundary condition at the inlet was the transferred velocity profile. An opening with ambient pressure was defined as the boundary condition at the flute. A rotational speed of 796 min^{-1} was applied to the drill and the fluid volume, which corresponds to a cutting speed of 20 m/min. The convergence criterion was 10^{-5}. In order to consider the heat input into the drill during the cutting process, an approach from Lazoglu et al [11] to simulate temperature fields on the drilling tool was adopted.

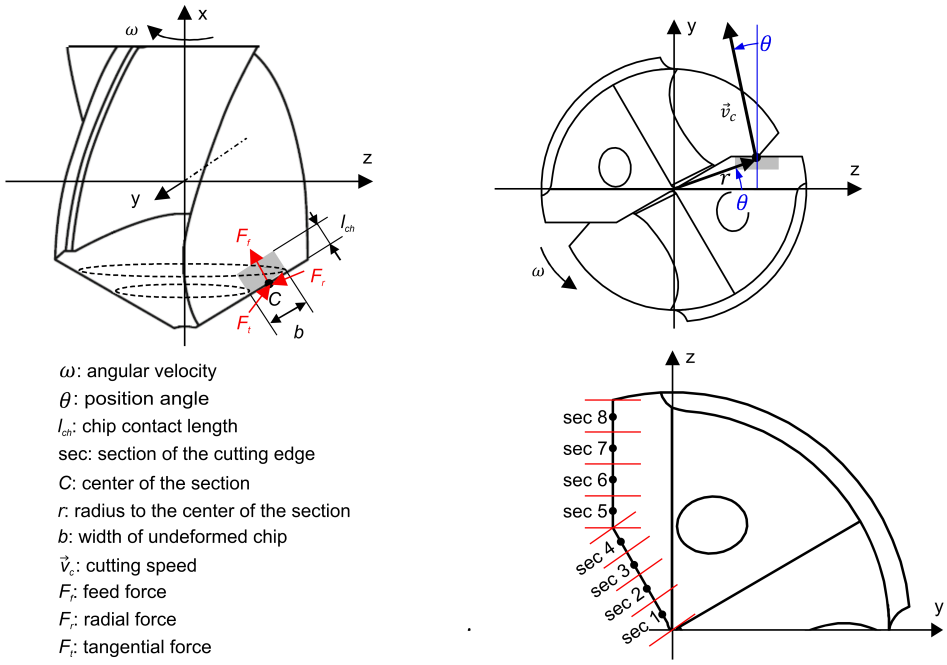


Figure 4 Discretization of the cross-section of the undeformed chip.

Thereby the heat input is estimated via the frictional power within the cutting process. For this purpose, the cutting forces along the main cutting edge are calculated using the discretized orthogonal to oblique transformation model as described by Altintas [1].The cross-section of the undeformed chip was discretized into eight sections (sec) according to Figure 4 and the discrete force components along the tangential F_t, feed F_f and radial F_r direction as well as the resultant cutting force F_{res} were calculated in every section according to the equations 2-5 [1]:

$$F_t = \frac{\tau_s bh[cos(\theta_n) + tan(\theta_i)tan(i)]}{[cos(\theta_n + \phi_n)cos(\phi_i) + tan(\theta_i)sin(\phi_i)]sin(\phi_n)} \tag{2}$$

$$F_f = \frac{\tau_s bhsin(\theta_n)}{[cos(\theta_n + \phi_n)cos(\phi_i) + tan(\theta_i)sin(\phi_i)]cos(i)sin(\phi_n)} \tag{3}$$

$$F_r = \frac{\tau_s bh[tan(\theta_i) - cos(\theta_n)tan(i)]}{[cos(\theta_n + \phi_n)cos(\phi_i) + tan(\theta_i)sin(\phi_i)]sin(\phi_n)} \tag{4}$$

$$F_{res} = \sqrt{F_t^2 + F_f^2 + F_r^2} \tag{5}$$

The tool orthogonal rake angle γ_n and the inclination angle i were determined on the geometry model for each segment. The friction angle β_α was calculated using the chip compression ratio r_c. The chip compression ratio r_c is the ratio of the uncut chip thickness h over the deformed chip thickness h_c [1]. The deformed chip thickness was determined experimentally ($h_c = 0.06$ mm).

$$i = sin^{-1}(sin(\theta)\ sin(\frac{\sigma}{2})) \tag{6}$$

$$\beta_\alpha = 45° + \gamma_n - tan^{-1}(\frac{r_c sin(\gamma_n)}{1 - r_c sin(\gamma_n)}) \tag{7}$$

$$r_c = \frac{h}{h_c} \tag{8}$$

The chip flow angle η, oblique shear angle ϕ_i, normal shear angle ϕ_n and the direction of the resultant force vector (θ_i and θ_n) were determined for each segment according to the "maximum shear stress principle" (Figure 5) [1]. The convergence criterion for the computation of the angles was chosen to be 10^{-12}.

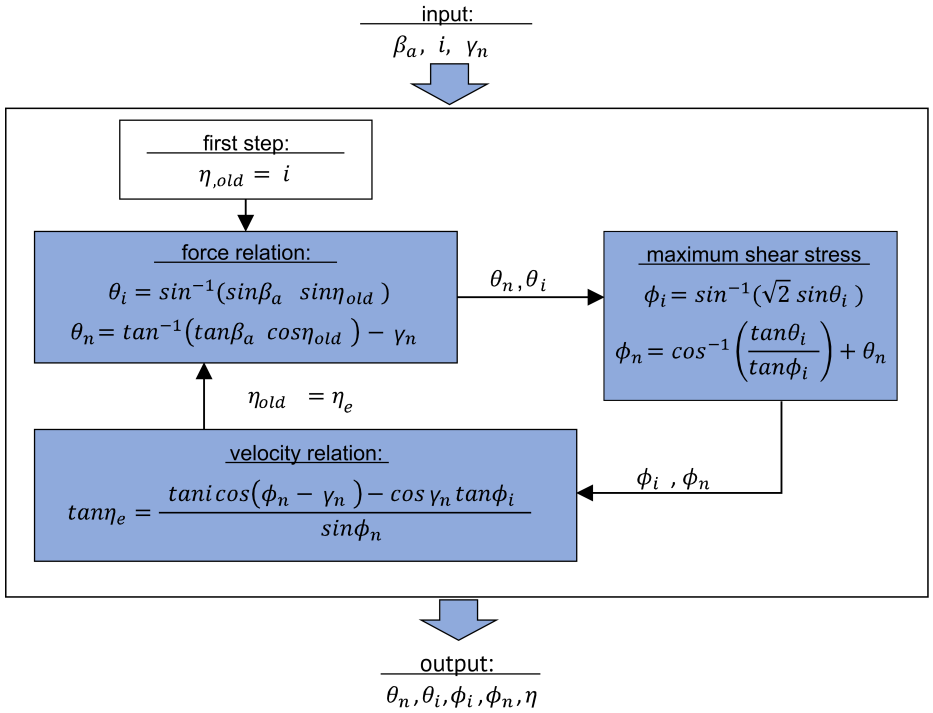


Figure 5 Calculation of the required angles according to the maximum shear stress principle.

The friction force F_{fr} in each segment can be determined using the friction angle β_α and the resulting cutting force F_{res} [1]:

$$F_{fr} = F_{res} sin(\beta_\alpha) \tag{9}$$

The resulting angles and forces along the main cutting edge are shown in Figure 6.

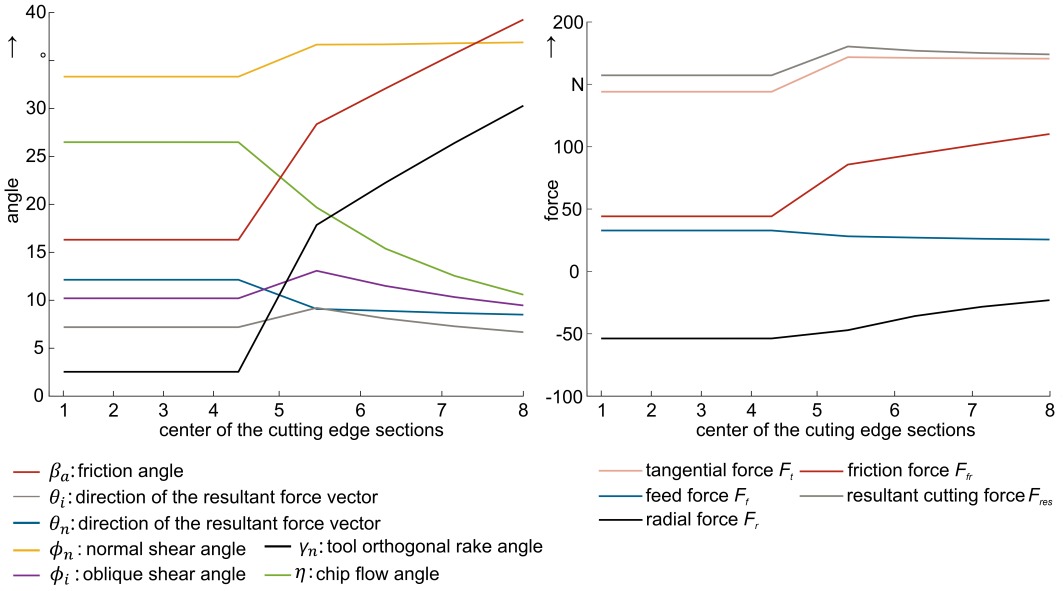


Figure 6 Calculated angles and forces along the main cutting edge.

The friction power P_{fr} is the product of the friction force F_{fr} and the chip velocity v_{ch}, given by [1]:

$$P_{fr} = F_{fr} v_{ch} \tag{10}$$

$$v_{ch} = \frac{v_c \sin(i) \sin(\Phi_n) \sec(\eta)}{\tan(\Phi_i) \cos(\gamma_n) + \sin(\Phi_n) \tan(\eta)} \tag{11}$$

The effective heat input P_{eff} into the drill for each section was obtained by including the heat partition coefficient λ [11]:

$$P_{eff} = \lambda P_{fr} \tag{12}$$

The heat partition coefficient λ was calculated corresponding to cutting speed v_{ch} of each section as reported in [18]:

$$\lambda = 1.17 \; {v_c}^{-0.293} \tag{13}$$

The effective heat input was introduced at the rake face in the area of the chip contact. For this purpose, the effective heat flux density q_{eff} was calculated for each segment of the rake face from the effective power P_{eff} and the surface A of the respective element.

$$q_{eff} = \frac{P_{eff}}{A} \tag{14}$$

For the heat transfer between fluid and solid the conservative interface flux option was used at the Fluid-Solid interface and the Solid-Fluid interface. All remaining surfaces were defined as adiabatic. The cutting zone is cooled indirectly via the flank face. In order to calculate the heat distribution, the heat capacity, thermal conductivity and density of the cemented carbide and metalworking fluid were included in the simulation according to manufacturer's

specifications. The chip contact length l_{ch} along the main cutting edge was calculated for every section according to [1]:

$$l_{ch} = \frac{h \sec(\eta) \sin(\Phi_n + \theta_n)}{\sin(\Phi_n)[\cos(\gamma_n)\cos(\theta_n) - \sin(\gamma_n)\sin(\theta_n)]} \tag{15}$$

The calculated heat input for each section is shown in Figure 7a. The heat input increases in the direction of the cutting edge corner (Figure 7b). This results in particular from the increase of the cutting speed along the main cutting edge in the direction of the cutting edge corner.

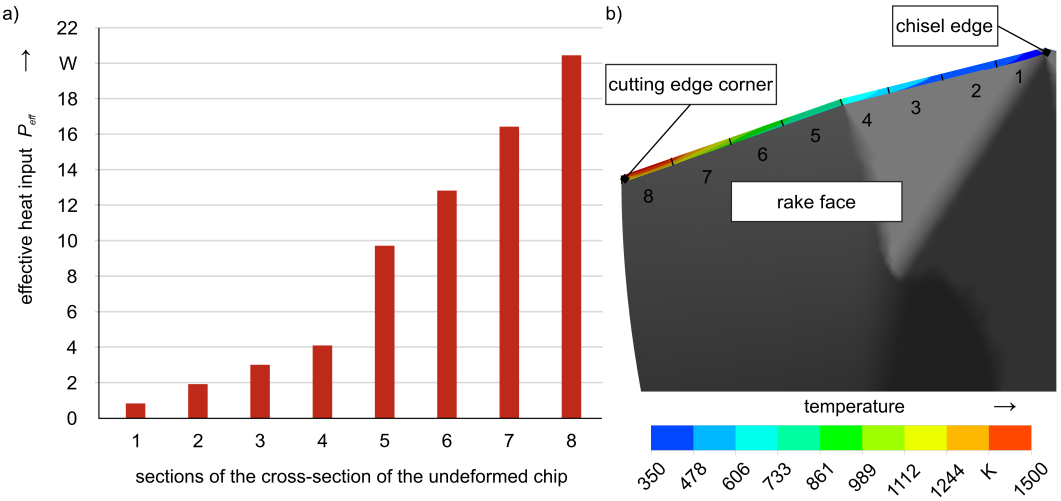


Figure 7 a) Effective heat input for every section b) Temperature distribution in the area of the chip contact length.

4 Results

4.1 Simulation

The flow velocities in the area of the first, second, and third flank face are shown as vectors in Figure 8. The vectors of the reference drill without structuring (ref.), show the highest flow velocities in the area of the second flank face. The flow velocities decrease in the area of the first flank face and in direction to the main cutting edge. Moreover, in some areas near the main cutting edge, dead zones can even be seen, i.e. areas with almost no flow velocity. This is in particular true in front of the cooling channel outlet and close to the cutting edge corner. By integrating the groove-shaped structures (S1 – S3) and the structure S4, the velocity can be increased in the area of the first flank face compared to the reference. The size of the dead zones can also be reduced, especially in front of the cooling channel outlet. In addition, the flow velocity is increased through the structuring in the area of the cutting edge corner. Especially the groove-shaped structures (S1 – S3) reduce the dead zones near the cutting edge corner. Comparing the groove-shaped structures (S1 - S3) with each other, there are no significant differences in flow distribution and flow velocities between S2 and S3. In the area of the additional slot between the cooling channel and the structure, S2 and S3 have slightly lower flow velocities than S1. Structure S4 shows the most even flow distribution in the area of the first flank face compared to the other drills. In particular, the dead zone in front of the cooling channel outlet is reduced compared to the reference, and also the dead zone close to the cutting edge corner is reduced.

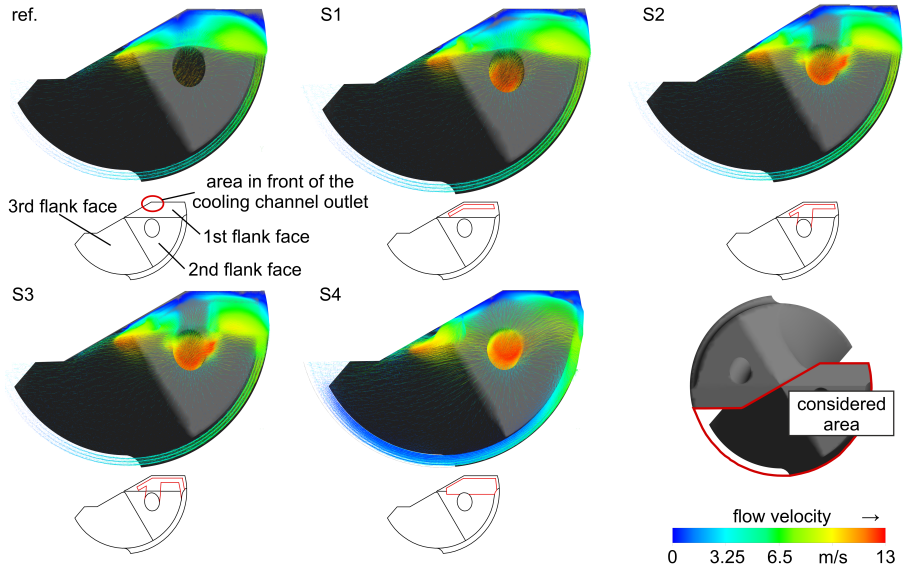


Figure 8 Resulting flow vectors at a volume flow rate of 1 l/min.

Previous investigations, dealing with the optimization of the cooling of internally cooled drills, have shown that an increase in the flow velocity near the main cutting edge and in the area of the first flank face lead to a reduction of the temperatures in the cutting wedge [3, 12, 16].

Figure 9 (left) shows the resulting temperatures in the area between the main cutting edge and the structures. The average temperature of the main cutting edge of all drills is given on the right hand side of Figure 9.

Compared to the reference, all structures cause a reduction in temperature, especially near the cutting edge corner. By integrating structure S1, the temperatures are reduced most significantly in the area of the cutting edge corner. In contrast to the reference drill, the average temperature of the main cutting edge is reduced by about 47 K. In comparison with the structure S1 and S3, structure S2 reduces the temperatures near the chisel edge most significantly, whereas the temperatures near the cutting edge corner are slightly higher compared to S1. Compared to the reference the average temperature of the main cutting edge is reduced by about 41 K. The temperature distribution of S3 is similar to that of S1, deviating only near the cutting edge corner with slightly higher temperatures. The average temperature of the main cutting edge compared to the reference is reduced by about 44 K for S3. The most significant reduction in the average temperature of the main cutting edge (54 K) compared to the reference is offered by structure S4. Furthermore, S4 has the lowest temperatures in front of the cooling channel outlet. However, higher temperatures occur near the cutting edge corner than for S1.

The combination of an increase in surface area and in fluid volume as well as overall improved flow conditions leads to an enhanced cooling of the drill. The described flow conditions above the first flank face are in-line with the resulting temperatures.

The resulting temperatures in the cutting wedge are shown on a plane 60 µm below the first flank face (Figure 10).

For all drill variants, the highest temperatures occur near the rake face and in the direction of the cutting edge corner. The reference drill (ref.) shows the highest temperatures of all drill variants on the entire plane. The groove-shaped structures as well as the structure S4

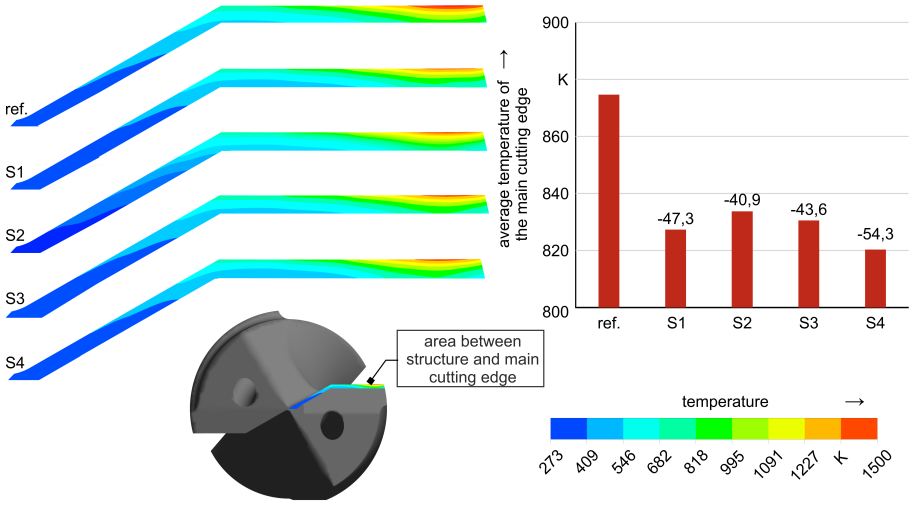


Figure 9 Resulting temperatures near the main cutting edge (left) and average temperature of the main cutting edge (right) at a volume flow rate of 1 l/min.

reduce the temperatures in particular near the rake face and in the area of the cutting edge corner. Structure S1 shows the lowest overall temperatures and leads to significantly lower temperatures near the cutting edge corner compared to the reference (ref.). S2, S3 and S4 have similar temperature distributions along the rake face.

4.2 Experiment

The results of the maximum width of flank wear land (VB_{max}) over the drilling path are shown in Figure 11. A degressive wear progress can be determined for all drill variants. Up to a drilling path of 0.3 m the maximum width of flank wear land increases strongly, thereafter the curve is characterized by a quasi-linear progress. The wear curves of the investigated drill variants start differing significantly after reaching a drilling path of 1.4 m. This is followed by a widening spread, where the structured drills (S1 - S4) show overall lower values of the maximum width of flank wear land in comparison to the reference drill (ref.) The most significant increase in tool life is caused by structure S1, with an improvement of about 21 % compared to the reference. Structures S2 and S3 lead to an increase in tool life of 11 % and 17 % respectively. S4 increases the tool life by 13 %. Since the same process conditions (feed rate, cutting speed, MWF flow rate and pressure) and the same drill geometry were used for all investigated drills, these longer tool lifetimes can only be attributed to the improved flow conditions resulting from the structuring.

The torques in the rotational axis and the forces in the feed direction in function of the wear condition and structuring are shown in Figure 12. An increase due to wear can be observed for all torque values. These increases compared to the unworn condition vary between 7 % to 10 %. For both wear conditions, no significant differences (max. 2 %) in torque can be identified in dependence of the structures. Furthermore, no systematic and no significant influence (max. 3 %) of the wear condition on the feed force can be identified. Also, no significant influence of the structuring and structural shape on the feed force (max. 4 %) can be detected.

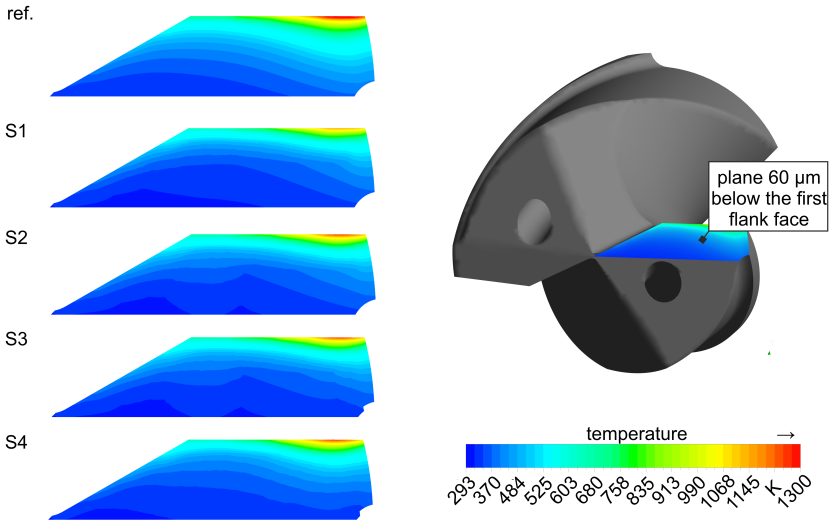


Figure 10 Resulting temperatures near the main cutting edge (left) and average temperature of the main cutting edge (right) at a volume flow rate of 1 l/min.

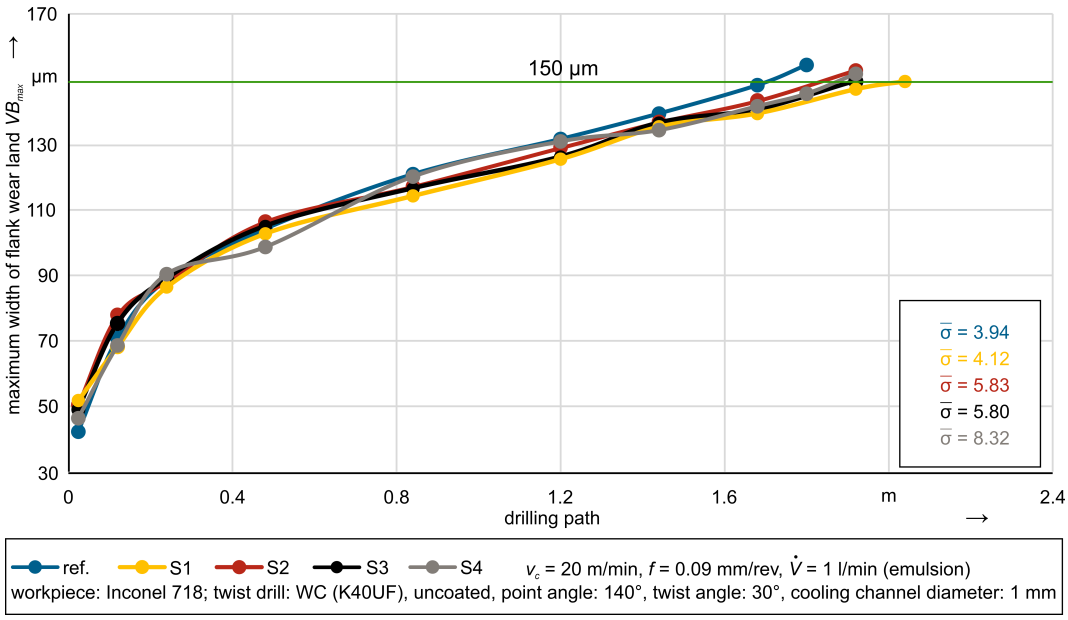


Figure 11 Maximum width of flank wear land (VB_{max}) depending on the structuring of the drills.

Figure 13 shows the surface roughness R_z and R_{max} of the borehole wall depending on the wear condition and structuring.

In the case of the structured drills, the graphs reveal that the surface roughness (R_z and R_{max}) of the borehole wall increases with increasing wear. The reference tool shows no differences for both wear conditions. The structured tools tend to provide a better surface roughness compared to the reference in unworn condition. S1 shows the smallest differences in surface roughness compared to the reference. Without wear, R_z and R_{max} are reduced by approx. 1 µm compared to the reference, at $VB_{max} = 150$ µm there are no differences. S2 has the lowest values for R_z and R_{max} at $VB_{max} = 0$ µm, but the values are about 1 µm

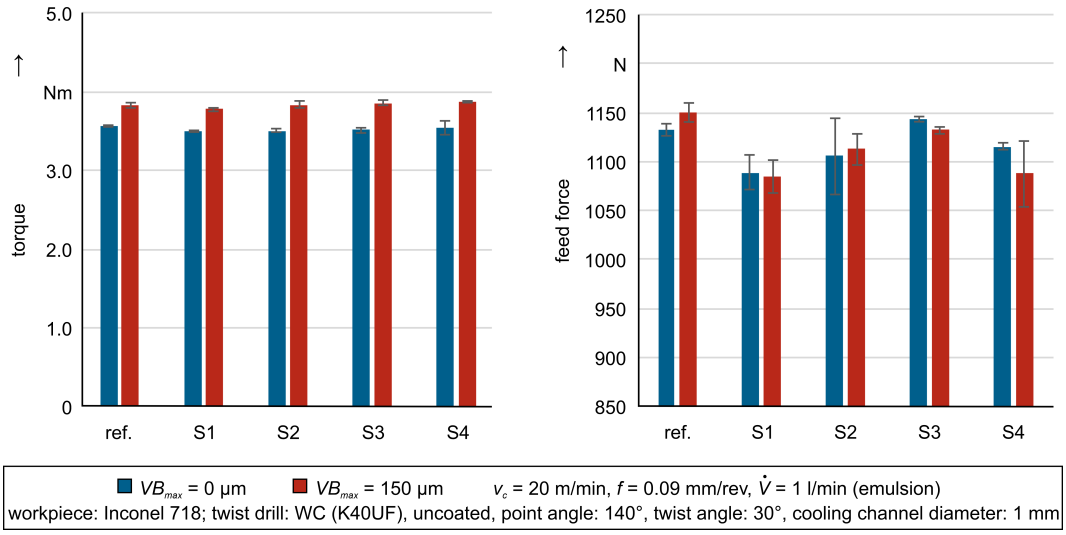


Figure 12 Measured torques in the rotational axis and forces in feed direction depending on the wear condition and structuring.

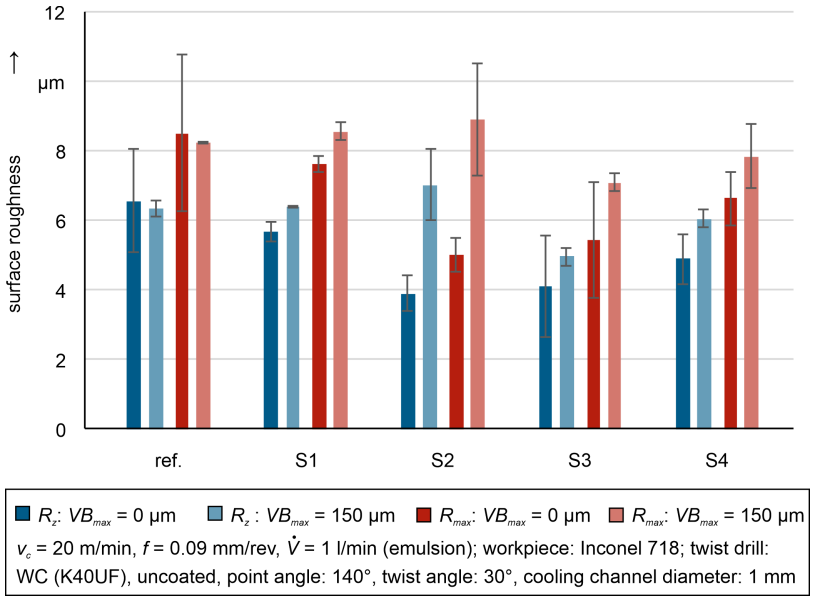


Figure 13 Measured torques in the rotational axis and forces in feed direction depending on the wear condition and structuring.

above the reference values in the worn state. For both wear conditions, S3 shows reduced surface roughness values (R_z and R_{max}) compared to the reference. Without wear, the values forR_z are about 2.5 µm below the values of the reference, at $VB_{max} = 150$ µm about 1.5 µm. In comparison with the reference, S4 has lower R_z and R_{max} values for both wear conditions. The differences are within a maximum range of 1.5 µm.

The borehole quality, determined by the evaluation of the roundness, concentricity and the borehole diameter, depending on wear as well as the structuring, is shown in Figure 14.

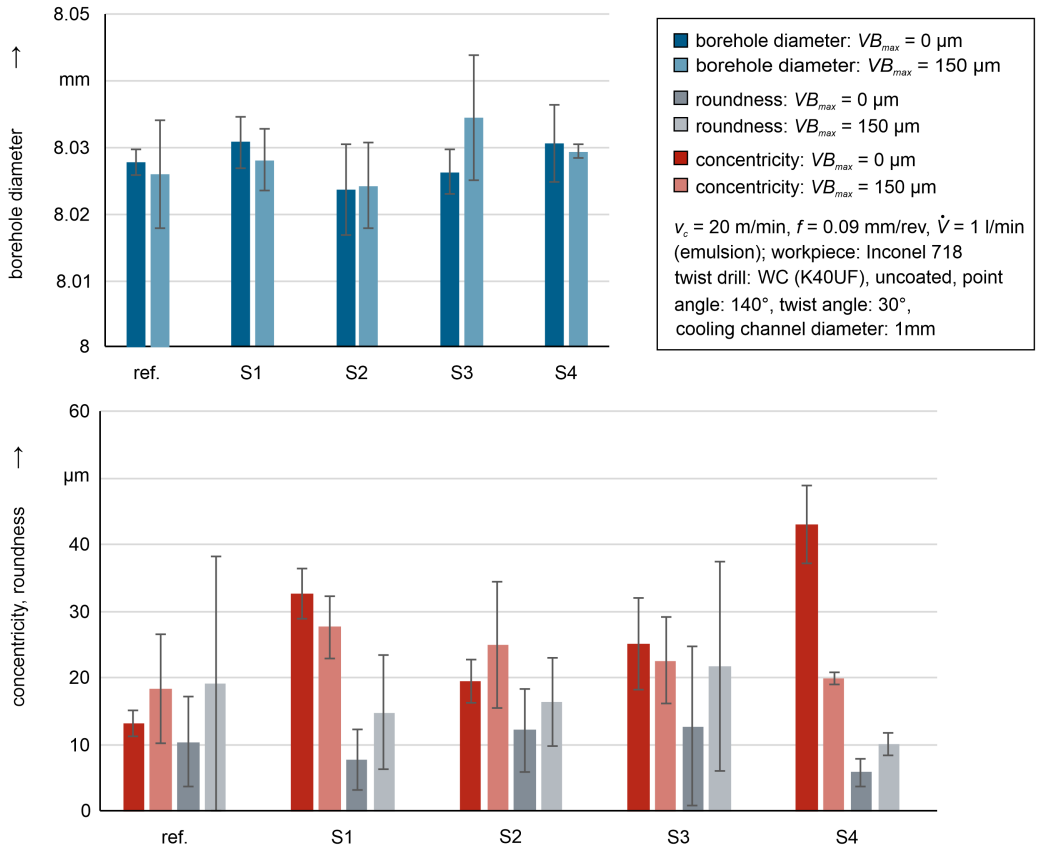


Figure 14 Measured torques in the rotational axis and forces in feed direction depending on the wear condition and structuring.

The measurements do not reveal any significant or systematic influence of the structuring and wear in terms of the diameter of the boreholes. The roundness values tend to increase with increasing wear. S4 shows an improved borehole roundness compared to the reference and other structured drills. The deviation in concentricity of S4, on the other hand, is significantly increased compared to the other drills for the unworn condition. At $VB_{max} = 150$ µm, however, the concentricity does not significantly differ from those of the other drill variants.

5 Conclusion and Outlook

In this paper the influence of different structured flank faces on cooling efficiency, tool life, process forces and borehole quality were investigated. For this purpose, the work was divided into a simulative and an experimental part. The aim of the simulative investigations was to analyze the influence of structures in the area of the first flank face on the flow conditions and thus the cooling of the drills. By means of the experimental investigations the interactions between optimized flow conditions and the tool life as well as the borehole quality could be investigated. The following conclusions can be drawn from the investigations:

The unstructured tool displays low flow velocities and larger dead zones in the area of the cutting edge. As a result, the cutting edge is not sufficiently cooled and is therefore exposed to high thermal loads. By integrating structures near the main cutting edge, the

flow conditions in this area could be improved. In particular a grooved structure along the main cutting edge reduces dead zones in this area and near to the cutting edge corner. As a result, the cooling is improved and the temperatures in the cutting wedge, especially near the thermally high loaded cutting edge corner could be reduced.

The results of the simulations show a good accordance with the conducted tool life tests. A correlation between simulated thermal load of the tools and the achieved tool life was shown. Compared to the non-structured drills, significant increases in tool life of up to 21 % have been achieved through the structuring. Since the same process conditions (feed rate, cutting speed, MWF-flow rate and pressure) and the same drill geometry were used for all investigated drills, these longer tool lifetimes can be attributed to the improved flow conditions resulting from the structuring.

The examined structures have no influence on the process forces. The observed differences in borehole quality should not be regarded as insignificant, as the functionality of the boreholes is not affected by any of the modifications.

In further studies, the influence of the structure depth and the distance to the main cutting edge will be investigated. In addition, the influence of the cooling channel design on the flow conditions of the MWF in the borehole will be investigated through visualization investigations using particle tracking.

[1] Naming of specific manufacturers is done solely for the sake of completeness and does not necessarily imply an endorsement of the named companies nor that the products are necessarily the best for the purpose.

References

1 Y. Altintas. *Manufacturing automation: metal cutting mechanics, machine tool vibrations, and CNC design*. Cambridge University Press, 2012.

2 Ansys. Cfx - pre user's guide, 2011.

3 N. Beer, E. Oezkaya, and D. Biermann. Drilling of inconel 718 with geometry-modified twist drills. *Procedia Cirp*, 24:49–55, 2014. doi:10.1016/j.procir.2014.07.124.

4 I.A. Choudhury and M.A. El-Baradie. Machinability of nickel-base super alloys: a general review. *Journal of Materials Processing Technology*, 77(1-3):278–284, 1998. doi:10.1016/S0924-0136(97)00429-9.

5 F. Fallenstein and J.C. Aurich. Cfd based investigation on internal cooling of twist drills. *Procedia CIRP*, 14:293–298, 2014. doi:10.1016/j.procir.2014.03.112.

6 F. Fallenstein and J.C. Aurich. Kühlkanalaustrittsbedingungen beim bohren. *wt Werkstattstechnik online*, 105:495–500, 2015.

7 D. Guo, X. Guo, K. Zhang, Y. Chen, C. Zhou, and L. Gai. Improving cutting performance of carbide twist drill combined internal cooling and micro-groove textures in high-speed drilling ti6al4v. *The International Journal of Advanced Manufacturing Technology*, 100(1-4):381–389, 2018. doi:10.1007/s00170-018-2733-z.

8 C. Hirsch. *Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics*. Elsevier, 2007. doi:10.1016/B978-0-7506-6594-0.X5037-1.

9 P. Hänle and M. Schwenck. Optimization of cutting tools using ca-technologies. *International Workshop on Modeling of Machining Operations*, 8:463–468, 2005.

10 F. Klocke. *Manufacturing Processes 1*. Springer-Verlag Berlin Heidelberg, 2011.

11 I. Lazoglu, G. Poulachon, C. Ramirez, M. Akmal, B. Marcon, F. Rossi, J.C. Outeiro, and M. Krebs. Thermal analysis in ti-6al-4v drilling. *CIRP Annals*, 66(1):105–108, 2017.

12 D. Mueller, B. Kirsch, and J.C. Aurich. The influence of structured flank faces on cooling performance when drilling. *Procedia CIRP*, 82:415–420, 2019. doi:10.1016/j.procir.2019.03.274.

13 D. Mueller, B. Kirsch, and J.C. Aurich. Kühlkanalaustrittsbedingungen bei bohrern. *wt Werkstattstechnik online*, 109(1/2):30–34, 2019.

14 D. Mueller, A. Lange, B. Kirsch, and J.C. Aurich. Kühlkanalaustrittsbedingungen beim bohren. *ZWF*, 113:471–474, 2018.

15 E. Oezkaya, N. Beer, and D. Biermann. Experimental studies and cfd simulation of the internal cooling conditions when drilling inconel 718. *International Journal of Machine Tools and Manufacture*, 108:52–65, 2016. doi:10.1016/j.ijmachtools.2016.06.003.

16 E. Oezkaya, M. Bücker, and D. Biermann. Simulative analyses focused on the changes in cutting fluid supply of twist drills with a modified flank face geometry. *International Journal of Mechanical Sciences*, 180, 2020. doi:10.1016/j.ijmecsci.2020.105650.

17 E. Oezkaya, I. Iovkov, and D. Biermann. Fluid structure interaction (fsi) modelling of deep hole twist drilling with internal cutting fluid supply. *CIRP Annals*, 68(1):81–84, 2019. doi:10.1016/j.cirp.2019.03.003.

18 J. Rech, P.J. Arrazola, C. Claudin, C. Courbon, F. Pusavec, and J. Kopac. Characterisation of friction and heat partition coefficients at the tool-work material interface in cutting. *CIRP Annals*, 62(1):79–82, 2013. doi:10.1016/j.cirp.2013.03.099.

19 S. Rodriguez. *Applied Computational Fluid Dynamics and Turbulence Modeling*. Springer, 2019.

20 H.K. Toenshoff and B. Denkena. *Basics of cutting and abrasive processes*. Springer-Verlag Berlin Heidelberg, 2013. doi:10.1007/978-3-642-33257-9.

21 D. Zhu, X. Zhang, and H. Ding. Tool wear characteristics in machining of nickel-based superalloys. *International Journal of Machine Tools and Manufacture*, 64:60–77, 2013. doi:10.1016/j.ijmachtools.2012.08.001.

Determination of Aggregate Elastic Properties of Powder-Beds in Additive Manufacturing Using Convolutional Neural Networks

Ardalan R. Sofi
Department of Mechanical and Aerospace Engineering, University of California at Davis, CA, USA

Bahram Ravani[1]
Department of Mechanical and Aerospace Engineering, University of California at Davis, CA, USA

Abstract

The most popular strategy for the estimation of effective elastic properties of powder-beds in Additively Manufactured structures (AM structures) is through either the Finite Element Method (FEM) or the Discrete Element Method (DEM). Both of these techniques, however, are computationally expensive for practical applications. This paper presents a novel Convolutional Neural Network (CNN) regression approach to estimate the effective elastic properties of powder-beds in AM structures. In this approach, the time-consuming DEM is used for CNN training purposes and not at run time. The DEM is used to model the interactions of powder particles and to evaluate the macro-level continuum-mechanical state variables (volume average of stress and strain). For the Neural Network training purposes, the DEM code creates a dataset, including hundreds of AM structures with their corresponding mechanical properties. The approach utilizes methods from deep learning to train a CNN capable of reducing the computational time needed to predict the effective elastic properties of the aggregate. The saving in computational time could reach 99.9995% compared to DEM, and on average, the difference in predicted effective elastic properties between the DEM code and trained CNN is less than 4%. The resulting sub-second level computational time can be considered as a step towards the development of a near real-time process control system capable of predicting the effective elastic properties of the aggregate at any given stage of the manufacturing process.

2012 ACM Subject Classification Applied computing → Industry and manufacturing

Keywords and phrases Additive Manufacturing, Convolutional Neural Network, Homogenization, Discrete Element Method, Powder-Bed

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.8

Funding This work was partially supported by a Space Technology Research Institutes grant from NASA's Space Technology Research Grants Program.

1 Introduction

Over the past few decades, additive manufacturing (AM) has become one of the mainstream manufacturing processes. Unlike the conventional subtractive manufacturing methods, AM is based on a layer-wise transformation of materials into the three-dimensional workpiece; therefore, it does not require fixtures, cutting tools, or other specialized tooling equipment. One of the most rapidly growing AM technologies to manufacture complex metallic and ceramic structures is Selective Laser Sintering (SLS), where a high-power laser fuses small powders into a desired three-dimensional (3D) shape. Physical modeling of powder-based AM structures is challenging due to the discrete nature of their structures. Several researchers modeled the powder bed as a continuum structure using FEM. However, as the number of required elements for particle-level modeling of large discrete structures increases, the

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 8; pp. 8:1–8:17

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

FEM becomes more computationally expensive. We modeled the bounded pair of powders in an AM structure as truncated spherical particles with elastic bonding. Every elastic-bound constitutes a 3D beam element between the centers of two powders; therefore, the manufactured aggregate constructs a space frame structure. An explicit expression of the symmetric stiffness matrix of the beam element is derived, and the force-displacement behavior of AM structure is modeled using the DEM. The macro-level volume average of the stress and strain tensor is found based on micro-level variables such as grain displacements and local geometrical characteristics. Therefore, the developed DEM code could be utilized to estimate the effective elastic properties of any powder-based AM parts. An alternative approach to such a homogenization method is to extract the appropriate set of patterns from the pixelated structure by convolving a weighted filter across the whole structure. Using the Convolutional Neural Network (CNN) technique, these features are then combined by the subsequent layers of CNN to detect higher-order features. The last fully-connected layer at the end of the network might be used to predict the elastic material properties of aggregate.

The effective implementation of powder-based AM technologies relies on the characterization of the final product based on manufacturing process parameters. These parameters include layer thickness, scanning strategy, and powder size and distribution [5]. Miranda et al. [23] studied the influence of laser scanning speed on the shear strength of stainless steel powder and developed a predictive regression model based on the manufacturing process parameters. Song et al. [35] investigated the effect of Selective Laser Melting (SLM) manufacturing process parameters on the microstructure of Ti6Al4V parts. Read et al. [30] used a statistical model to develop the experimental design for investigation of the effect of process parameters on the porosity formation in powder-based AM parts. Calignano et al. [4] used statistical techniques to study the relationship between surface roughness of parts produced by Direct Metal Laser Sintering (DMLS) and AM process parameters. Calignano et al. showed that scanning speed has a significant influence on the surface roughness of the final product. All mentioned experimental studies are expensive, time-consuming, and could be used to predict only specific properties of the final product based on the change of some particular process parameters.

FEM is the most largely used method for simulation of the thermomechanical behavior of powder beds at the scale of the part geometry. Using the SIMULIA simulation with a relatively large element size (0.2 mm), Yang et al. [40] studied the remained elastic strain in cantilever structures. Gu and He [10] developed an FEA model to evaluate the residual stresses of SLM manufactured parts. Gu and He concluded that the maximum residual stress in SLM parts was located at the end of the first and last track. Singh and Srinivasa [33] developed an FEA model to predict the density distribution in a single layer of powder and optimized process parameters for wanted density. Generally, the continuum model of powder bed using FEA is relatively faster than discrete models, although these continuum models are not capable of generating valid results for microstructural behavior of AM structures [8]. In order to analyze the microstructural behavior of AM structures, the mesh size for continuum models must be smaller than the size of particles. Hence, the microstructural molding of powder-based AM structures with FEA needs a large number of mesh elements; consequently, the FEA becomes even more computationally expensive than discrete models.

Developing a fast and accurate simulation tool for powder-based AM structures is challenging due to the stochastic nature of powdered material [9]. Therefore, a particle-based numerical model is required for the physical modeling of powder interaction during the additive manufacturing process [26]. One of the most accurate numerical techniques to study the mechanics of particle interactions is DEM [37]. Using DEM simulations, Haeri et al. [11] studied the influence of powder bed thickness and velocity of the powder spreading

devices on the surface roughness of the final AM structure. The DEM could even be used for elasticity analysis of continuum structures. For instance, Gao et al. [6] transferred the continuum structure into a discrete model and utilized the DEM model to analyze the force and displacement in the discretized structure.

Relating the macroscopic mechanical properties of AM parts to their microscopic structural properties is the essential goal of physical modeling of the powder-based additive manufacturing processes. The most popular strategy for the homogenization of material properties in AM structures is through FEM or DEM numerical solutions; however, these well-known homogenization techniques are time-consuming. Therefore, the homogenization of material properties in AM parts using FEM or DEM numerical techniques could limit our ability to develop a simulation-based real-time control system to improve the quality of the final product. Gobal and Ravani [8] reduced the computational time needed to simulate the thermo-mechanical behavior of the SLS manufactured structured by introducing the Adaptive Discrete Element Model (ADEM), where the size of discrete elements inside the powder bed increases in the areas far from the laser heat source. Liu and Shapiro [22] developed a mesoscale model of parts manufactured by a fused deposition process for homogenization of the elastic properties of AM structure [36]. Liu's homogenization technique computes the effective elasticity tensor of Fused Deposition Modeling (FDM) structures seven times faster than conventional finite element solvers. These homogenization methods are not fast enough to develop a simulation-based real-time control system for the AM process.

The Machine Learning (ML) methods, such as the fully-connected Artificial Neural Network (ANN) or Convolutional Neural Network (CNN), have been demonstrated to be valid ways to perform complex regression and pattern recognition for different manufacturing domains like process planning or production control [28]-[39]. Over the last decade, many studies have been done to use computer-vision hardware coupled with different neural network architecture to maximize performance in defect detection and defect classification in additive manufacturing process control. The interactions between the re-coater blade and powder bed during Laser Powder Bed Fusion (L-PBF) additive manufacturing process might cause re-coater hopping defect, super-elevation defect, etc. In order to autonomously detect spreading anomalies, Scime and Beuth [31] re-trained the already existing AlexNet CNN architecture using the transfer learning technique. Yuan et al. [41] set up a monitoring system and successfully applied semi-supervised CNN for in-situ monitoring of the SLM manufacturing process. In addition to the application of ML technique in computer-vision problems for AM, there has been limited work in the application of ML to predict the thermal behavior of AM structures (see [27]-[24]). To the best of our knowledge, in all the aforementioned works, there is no particle-level DEM simulation of powder-based AM structure combined with CNN algorithms to homogenize the effective elasticity tensor of powder-based AM structures.

2 Bounding Forces and Momentum of Partially Sintered Particle

The first step in the development of a DEM model to predict the effective elastic properties of powder-based AM structures for later training of the Neural Network is to evaluate the force-displacement behavior of every pair of sintered powder. Jefferson et al. [13] used FEA to derive the particle response rule for both normal and tangential relative deformation between every pair of particles and showed that this force-displacement model agrees with the Hertzian contact theory for small indentation. Then, Liu et al. [21] simplified Jefferson's formulation and modeled every pair of partially sintered powder as truncated elastic spheres overlapping neighboring particles with an elastic bridge that could transmit normal and tangential forces as well as rotational moments.

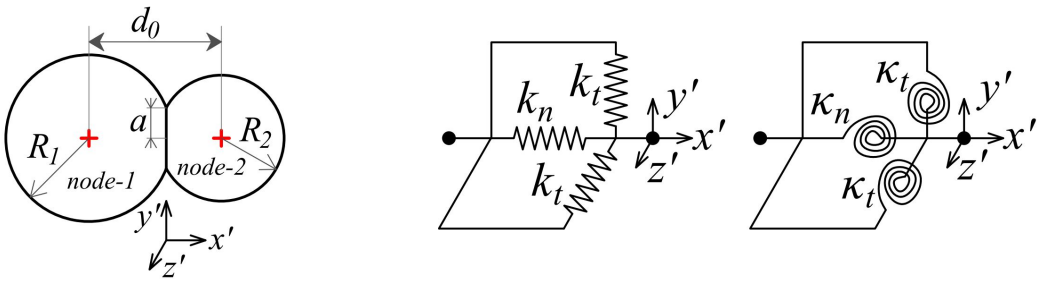


Figure 1 Schematic of a pair of sintered particles (left), and a system of two particles linked by six springs (right).

In this paper, the force-displacement between every pair of particles modeled using a single normal spring k_n, two identical tangential springs k_t, one torsional spring normal to the contact plane κ_n, and two identical torsional spring tangent to the contact plane κ_t (see Figure 1). Using the Liu et al. force-displacement model, Gobal [7] derived the following spring constants, which we used in our model:

$$k_n = \frac{Ea}{1-\nu^2}\left\{\frac{1+a^*\left[\frac{\pi}{6}\left(1-\nu^2\right)\left(1+2a^*\right)-a^*\right]}{\sqrt{\left(1-a^{*2}\right)}-\bar{\psi}\left(a^*+a^{*2}\left[\frac{\pi}{6}\left(1-\nu^2\right)\left(1+2a^*\right)-a^*\right]\right)}\right\} \tag{1}$$

$$k_t = \frac{2Ea}{(2-\nu)(1+\nu)}\left\{\frac{1+a^*\left[\frac{\pi}{6}\left(1-\nu^2\right)\left(1+2a^*\right)-a^*\right]}{\sqrt{\left(1-a^{*2}\right)}}\right\}, \tag{2}$$

$$\kappa_n = \frac{8ER^3a^{*3}}{(2-\nu)(1+\nu)}\left\{\frac{1+a^*\left[\frac{\pi}{6}\left(1-\nu^2\right)\left(1+2a^*\right)-a^*\right]}{\sqrt{\left(1-a^{*2}\right)}}\right\}, \tag{3}$$

$$\kappa_t = \frac{2ER^3a^{*3}}{(1-\nu^2)}\left\{\frac{1+a^*\left[\frac{\pi}{6}\left(1-\nu^2\right)\left(1+2a^*\right)-a^*\right]}{\sqrt{\left(1-a^{*2}\right)}-\bar{\psi}\left(a^*+a^{*2}\left[\frac{\pi}{6}\left(1-\nu^2\right)\left(1+2a^*\right)-a^*\right]\right)}\right\}, \tag{4}$$

where $a^* = a/R$ (a is the contact radius, $R^{-1} = R_1^{-1} + R_2^{-1}$, and $\bar{\psi}$ is a geometric factor that should be determined for each bond from the exact load distribution on the particle [13]. Liu et al. [21] tried several values of geometric factor and found $\bar{\psi} = 0.08$ to gave the best fit to the experimental data; therefore, We used the same value for our simulation. It is worth noting that a^* must be smaller than one ($a^* < 1$) to avoid the proposed stiffnesses in Eq. (1) to Eq. (4) turning into complex numbers.

3 Local and Global Stiffness Matrix for DEM Analysis of AM structure

Gao et al. [6] constructed a beam element within every two neighboring nodes with five degrees of freedom (DoF) per node and obtained a 10×10 symmetric stiffness matrix of the local equivalent beam element using the unit displacement method.Again Gobal [7] considered a 12 DoF force-displacement model ($\mathbf{f}' = \mathbf{K}'\mathbf{u}'$) which we are using here. This force-displacement model is between two particles where the displacement vector is

$\mathbf{u}' = (u'_1, u'_2, ..., u'_3)$ and force vector is $\mathbf{f}' = (f'_1, f'_2, ..., f'_3)$. Using the unit displacement method, Gobal [7] developed the local stiffness matrix as follows:

$$\mathbf{K}' = \begin{bmatrix} \mathbf{K'}_{11} & \mathbf{K'}_{12} \\ \mathbf{K'}_{21} & \mathbf{K'}_{22} \end{bmatrix}, \tag{5}$$

where

$$\mathbf{K'}_{11} = \begin{bmatrix} k_n & 0 & 0 & 0 & 0 & 0 \\ 0 & k_t & 0 & 0 & 0 & k_t d_0 \\ 0 & 0 & k_t & 0 & -k_t d_0 & 0 \\ 0 & 0 & 0 & \kappa_n & 0 & 0 \\ 0 & 0 & -k_t d_0 & 0 & \kappa_t + k_t d_0^2 & 0 \\ 0 & k_t d_0 & 0 & 0 & 0 & \kappa_t + k_t d_0^2 \end{bmatrix},$$

$$\mathbf{K'}_{12} = \begin{bmatrix} -k_n & 0 & 0 & 0 & 0 & 0 \\ 0 & -k_t & 0 & 0 & 0 & k_t d_0 \\ 0 & 0 & -k_t & 0 & -k_t d_0 & 0 \\ 0 & 0 & 0 & -\kappa_n & 0 & 0 \\ 0 & 0 & k_t d_0 & 0 & k_t d_0^2 - \kappa_t & 0 \\ 0 & -k_t d_0 & 0 & 0 & 0 & k_t d_0^2 - \kappa_t \end{bmatrix},$$

$$\mathbf{K'}_{21} = \begin{bmatrix} -k_n & 0 & 0 & 0 & 0 & 0 \\ 0 & -k_t & 0 & 0 & 0 & -k_t d_0 \\ 0 & 0 & -k_t & 0 & k_t d_0 & 0 \\ 0 & 0 & 0 & -\kappa_n & 0 & 0 \\ 0 & 0 & -k_t d_0 & 0 & k_t d_0^2 - \kappa_t & 0 \\ 0 & k_t d_0 & 0 & 0 & 0 & k_t d_0^2 - \kappa_t \end{bmatrix},$$

$$\mathbf{K'}_{22} = \begin{bmatrix} k_n & 0 & 0 & 0 & 0 & 0 \\ 0 & k_t & 0 & 0 & 0 & -k_t d_0 \\ 0 & 0 & k_t & 0 & k_t d_0 & 0 \\ 0 & 0 & 0 & \kappa_n & 0 & 0 \\ 0 & 0 & k_t d_0 & 0 & k_t d_0^2 + \kappa_t & 0 \\ 0 & -k_t d_0 & 0 & 0 & 0 & k_t d_0^2 + \kappa_t \end{bmatrix}.$$

The first step to assemble the global stiffness matrix is to perform the coordinate transformation on $\mathbf{K'}_{11}$, $\mathbf{K'}_{12}$, $\mathbf{K'}_{21}$, $\mathbf{K'}_{22}$ of every constructed beam element. The coordinate transformation of each block of the stiffness matrix could be done only on their tensorial form; therefore, the 6×6 blocks of stiffness matrix must first convert to $3 \times 3 \times 3 \times 3$ fourth-order tensorial form ($\mathrm{K'}_{ijkl}$). Each one of these fourth-order tensors must undergo coordinate transformation using the following equation:

$$\mathrm{K'}_{ijkl} = R_{im} R_{jn} R_{kp} R_{lq} \mathrm{K'}_{mnpq}, \quad \mathbf{R} = \mathbf{R}_z(\psi)\mathbf{R}_y(\phi)\mathbf{R}_x(\theta), \tag{6}$$

where $\mathbf{R}_x$, $\mathbf{R}_y$, and $\mathbf{R}_z$ are rotation matrices. After coordinate transformation, the global stiffness matrix could be obtained by assembling every block of the local stiffness matrix into their corresponding position in $6\mathrm{N} \times 6\mathrm{N}$ global stiffness matrix (N is the number of sintered particles). After subjecting every direction of every particle to either displacement or tractional boundary condition (BC), we could solve for displacement by inverting the global stiffness matrix.

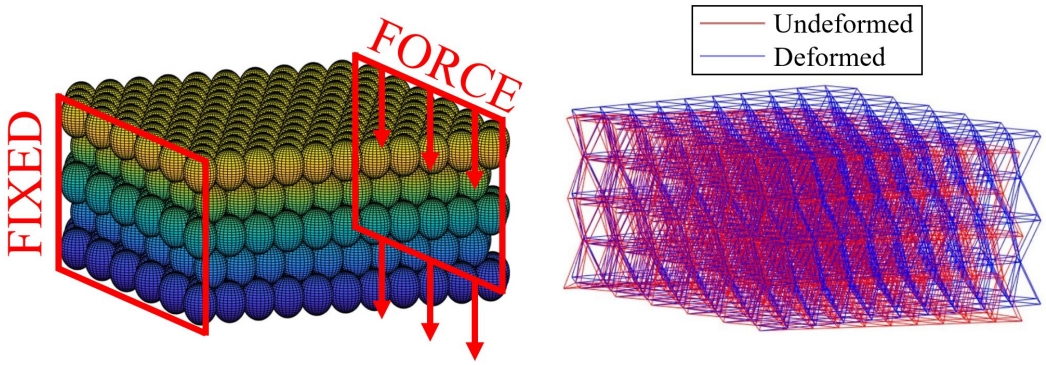


Figure 2 HCP structure for 0.25 mm × 0.25 mm × 0.1 mm AM part subjected to displacement and traction BC (left), deformed and undeformed generated spaceframe structure (right).

4 Simulation of Powder Bed Packing

In this section, we present a simple model of powder bed packing for elastic DEM analysis of powder-based AM structure. A number of researchers utilized Monte Carlo (MC) simulation to generate the random packing of unequal spherical particles [12]-[3]. The generated powder bed model by MC method was successfully used in the simulation of thermal behavior of AM processes (see [8]-[20]-[43]); Although these MC methods could not guarantee the structural determinacy of AM structure, they could generate a system of particles with complex stiffness values ($a^* > 1$). In this work, a simple Hexagonal Closest Packed (HCP) structure has been used to simulate the powder bed packing; consequently, the generated structure would be structurally determinant. To create a 3D HCP structure with unequal spherical particles, first, a completely packed 2D structure of uniform circular particle with radius lower than the desired average radius of particles generated then the second layer of powders could be generated by connecting the centers of the neighboring particles in the first layer and constructing 2D triangular elements. Consequently, the particles in the second layer can be generated at the center of each triangular element. The next layer of particles could be simply generated by copying the position of the particles from already generated first and second layers.

Each particle in the generated HCP structure just touching its neighboring particle, although these particles are not yet sintered together, and therefore the beam element could not be constructed between them. In order to create a realistic model of sintered particles, the radius of each particle in the HCP structure increased randomly between 5% to 14% of the radius of each particle. The increase in powder radii could not be set to more than 14% because it would cause the stiffnesses (Eq. (1) to Eq. (4)) to be complex numbers ($a^* > 1$). To ensure that this HCP model resembles the real sintered powder-based structure the position of each powder particle randomly changes within a small cubic box. The center of these cubic boxes is defined at the center of each undisplaced particles in the HCP structure, and each side of these boxes is $l = 2.05R_i$.

To model the geometry of the final product, a code was developed that first generates a fully packed HCP structure, then the powders that are not within the prescribed scanning path of the laser are deleted. Schematic of the laser scanning path on 3 mm × 3 mm domain. Figure 3 shows the scanning path on the AM structure, the red area is packed with particles, and there are no particles in the white space. The average radius of the spherical particle is set to be 12.5 μm.

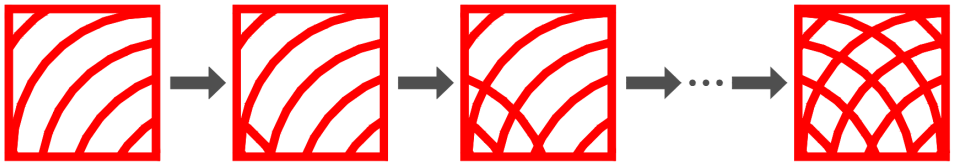


Figure 3 Schematic of the laser scanning path.

5 Microstructural Stress and Strain Tensor of Granular Assemblies

Because of the discrete nature of AM structure, the definition of continuum mechanical state variables stress and strain tensors are not self-evident. There are different approaches to find a theoretically correct definition of state variables based on grain displacements, local geometrical characteristics, and forces transmitted between contacting particles. In this work, we used the Katalin Bagi [2]-[1] definition of state variables in a granular assembly to obtain a volume average of stress and strain in the DEM simulation.

Bagi used the concept of *material cell* and *space cell* systems to discretize the unit normal vector in the Gauss-Ostrogradski equation and obtained the volume average of displacement gradient tensor as follows:

$$\bar{e}_{ij} = \frac{1}{V}\iiint\limits_{(V)} e_{ij}dV = \frac{1}{V}\iint\limits_{(S)} u_i n_j dS = \frac{1}{V}\sum_{m<n} \Delta u_i^{mn} d_j^{mn}, \tag{7}$$

where V is the volume of all constructed tetrahedral by beam elements and $\Delta u_i^{mn} = u_i^m - u_i^n$ is the relative displacement vector between every pair of particles obtained from already developed force-displacement DEM code, and d_j^{mn} is the *complimentary area vector* that can be defined as:

$$d_i^{mn} = \frac{1}{4}\sum_{t=1}^{T}\left(a_i^{m(t)} - a_i^{n(t)}\right), \quad a_i^k = -\frac{1}{3}b_i^k, \tag{8}$$

where particle (grain) G_m and G_n are sintered together, and the beam element is constructed between the centers of two particles. Now, all tetrahedral (space cells) that contain edge mn must be collected to calculate the complimentary area vector. In Eq. (1) T is the number of collected tetrahedral for edge mn and b_i^k is a vector corresponding with each four-face of every collected tetrahedron (the number of b_i^k vector for edge mn would be equal to $4T$). For instance, in Figure 4 three tetrahedral surrounding the edge $mn\,(T = 3)$. The magnitude of b_i^k vector is equal to the area of its corresponding face and the direction of this vector is normal to the face and pointing outward. The symmetric part of the displacement gradient would be the strain tensor.

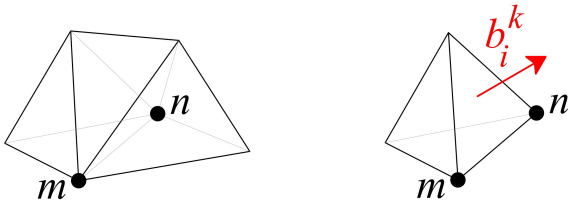


Figure 4 Schematic of three tetrahedral between six sintered particles (left), and the b_i^k (right).

Bagi expressed the Cauchy stress tensor for granular material with volume V as a summation of the dyadic product of contact forces between particles f_i^c and branch vector $(l_j^c = x_j^m x_j^n)$ as follows:

$$\bar{\sigma}_{ij} = \frac{1}{V} \iiint_{(V)} \sigma_{ij} dV = \frac{1}{V} \sum_{b \in B} f_i^b x_j^b = \frac{1}{V} \sum_{c \in V \cup B} f_i^c l_j^c, \tag{9}$$

where b is the number of particles subjected to external loading, and c is the number of contacts between powder particles. The effective elastic constants could be evaluated by performing six independent experiments for the same structure with different boundary conditions [22]. The effective compliance matrix $\mathbf{S}^{\text{eff}}$can be defined as follows:

$$\bar{\boldsymbol{\varepsilon}} = \mathbf{S}^{\text{eff}} \bar{\boldsymbol{\sigma}}. \tag{10}$$

The six general elastic properties (E_x, E_y, E_z, ν_{yz}, ν_{xz}, ν_{xy}) can be obtained from elements of the compliance matrix.

6 The Dataset and Pre-processing

The proposed method for the homogenization of elastic properties of powder-based AM structures in Sections 2 to 5 is not fast enough for developing a real-time control system capable of predicting the elastic properties of the aggregate at any given stage of manufacturing processes. In this section, an alternative approach to the homogenization of the powder-based AM structure presented using the CNN technique. The CNN that was initially introduced by Lecun et al. [16]-[18] has become an essential neural network architecture for computer vision tasks and image possessing. Local and global patterns of an image can be extracted with a convolutional layer; therefore, by nesting many convolutional layers in a hierarchical manner, CNN attempt to extract a broader structure from an image. The first step for implementing the CNN is to gather and prepare a dataset including hundreds of image representations of 3D printed structures and their corresponding elastic properties (E_x, E_y, E_z, ν_{yz}, ν_{xz}, ν_{xy}).

An image representation of powder-based AM structure could easily be obtained by discretizing the AM structure with small uniform cuboid boxes and calculating the fraction of the volume of each box filled with powders to the total volume of the box. For example, a structure with 3mm $\times$ 3mm base and three layers of powder could be discretized with 0.03mm $\times$ 0.03mm $\times$ 0.1mm cuboid; therefore, the structure would be turned into a 100×100 2D array of numbers (image) where each element of this array would be a number between 0 (empty cell) and 1 (fully-packed cell). Each one of these pixelated structures could be labeled with their corresponding elastic properties obtained with a time-consuming DEM homogenization process.

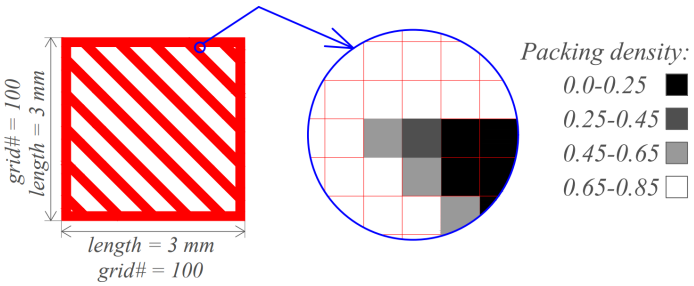


Figure 5 The pixelation process to present the AM structure with a 2D array of numbers.

7 Estimating the Effective Elastic Properties using CNN

CNN is an essential tool of deep learning, and it has demonstrated exceptional prediction performance within the field of computer vision [15]-[29]. CNN extracts features from images (2D or 3D arrays of numbers) and uses the backpropagation algorithm to optimize the learnable parameters in the network (weights and biases) [44]. Usually, CNN architecture contains multiple nested convolutional and pooling layers; subsequently, the data get flattened and passed into stack of fully connected layers [38].

The building blocks of convolutional layers are small learnable matrices called filters, which are used to extract the spatial features from the input array using the convolution operation. In order to create a complex decision boundary, the output of the convolutional operation would be passed into a nonlinear activation function. The typical activation function in a deep CNN architecture is the Rectified Linear Unit (ReLU) because of its short backpropagation's computational time compared to other nonlinear activation functions such as $tanh$ [19].

In order to reduce the number of extracted features and consequently decrease the number of learnable parameters, the output of the nonlinear activation function would be passed into a pooling layer. The most common form of pooling in CNN architectures is max pooling since it is extracting the most important features while reducing the size of the input. In order to extract meaningful data from pixel data, convolutional operation, activation function, and pooling operation must be stacked together [39].

Typically, the fully connected (FC) layers are the last part of CNN architectures [42]. All the neurons in the FC layers are connected to all the units of the previous layer. This full connection between input and output gives the model the ability to thoroughly mix the flow of information between the input and output of the FC layer; Therefore, the final output of CNN would be based on the whole image. The last FC layer (output layer) of conventional CNN architectures for multi-classification tasks would have a softmax activation function that computes the probability values of each class due to its mathematical definition. This means that by using the softmax activation function, the sum of the output values would be equal to 1 [14]. Although, in this work, we're trying to estimate the six independent, effective elastic properties of given pixelated AM structures; Hence, we modified the standard CNN architecture by replacing the classifier output layer with a multiple output linear regression layer. For a CNN with T outputs, where each output corespond to a distinct regression task, all T tasks share the same N input pixelated AM structures $\{x_i\}_{i=1}^{N}$, but have different class label $\{\{y_i^t\}_{i=1}^{N}\}_{t=1}^{T}$ [25]. Assuming every task (output) having same importance coefficient, the mean squared error loss function of the multiple output CNN regressor can be written as follows:

$$J(W,b) = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\left(y_i^t - \hat{y}_i^t\right)^2 \tag{11}$$

where $\hat{y}_i^t$ is the output of t-th regression task for i-th pixelated AM structures. Also, W and b are the weight and bias parameters of filters and FC layers for the entire CNN architecture that must be leaned using the backpropagation algorithm.

The output layer of several well-known deep CNN architectures for the multi-classification task (AlexNet [15], LeNet-5 [17], and VGG Net [32]) is replaced with the above proposed multiple output linear regression layer and used for training. These modified networks are backpropagated on proposed loss (error between predicted and actual effective elastic properties of AM structure) and optimized using an adaptive moment estimation (Adam)

algorithm. The original dataset is randomly split into 1600 training examples, and 400 of the original AM structures are used for validation purposes. All hyperparameters were kept as Keras API's default value except the batch size, which set to be 100 training examples for each iteration.

Training the modified AlexNet, LeNet-5, and VGG Net for 80 epochs results in training accuracy of 92.5%, 90.6%, and 18.1%, respectively. Despite the VGG Net's enormous number of trainable parameters (65.07 million parameters), its accuracy is very limited for our task. VGG Net's poor performance might be caused by vanishing gradient issues in very deep neural networks [34] or might be caused by the small number of training examples. The AlexNet with 20.31 million and LeNet-5 with only 59 thousand trainable parameters perform way better than the VGG Net. Although even with AlexNet's smaller number of parameters compared to VGG Net, the forward propagation process for an unseen AM structure might not be fast enough for a real-time control system. The LeNet-5 computational time for forward propagation is shorter than AlexNet, although it's less accurate. In order to achieve high accuracy and acceptable computational efficiency, we propose a new CNN architecture called Custom Net optimized for homogenization of aggregate elastic properties of powder-beds.

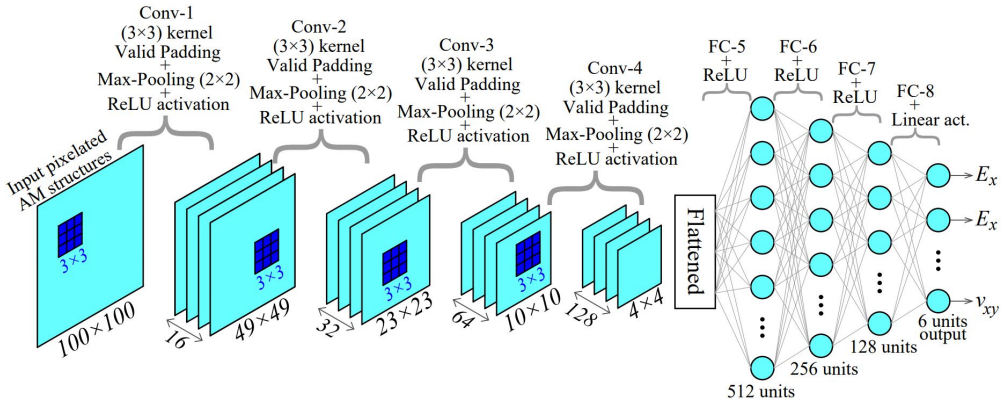


Figure 6 The CNN architecture of the proposed Custom Net.

As shown in Figure 6, the Custom Net consists of four stacks of convolutional operation, ReLU activation function, and pooling operation followed by four FC layers. The three hidden FC layer has ReLU activation function, and the last FC layer uses the multiple output linear regression layer. All convolutional (Conv) layer has the same 3×3 kernels, and the number of Channels gradually increases from 16 in Conv-1 to 128 in Conv-4. Every Max-Pooling operation has a 2×2 filter, and the number of units in FC layers reduces from 512 units in FC-5 to 128 units in FC-7. The Custom Net trained with the same optimizer and hyperparameters used for training of the three well-known deep CNN architectures.

The 1.31 million trainable parameters of the proposed Custom Net is significantly smaller than VGG Net and AlexNet's parameters. Therefore, Custom Net is easier to train and faster to implement for an unseen example compared to other architectures. Figure 7 illustrates the loss versus the number of batches of data that feed into all four CNN architectures, and one can see that the Custom Net converged faster than the other three CNN architecture, and after 1200 batch of training example Custom Net is more accurate than Alex Net and LeNet-5. The Training and validation accuracy of Custom Net over the increasing number of epochs (training iterations) are shown in Figure 8. After training the Custom Net for

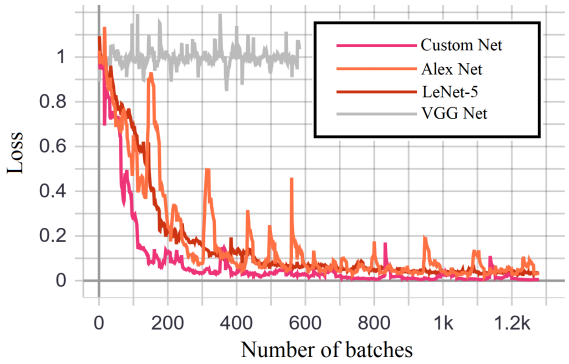


Figure 7 Loss (error) Vs. the number of batches of data for AlexNet, LeNet-5, VGG Net, and proposed Custom Net.

80 epochs, it's performing 96.5% accuracy on the training set and 96.1% accuracy on the validation set. The negligible difference between training and validation accuracy shows that this model has a low variance; therefore, the additional overfitting prevention strategies (dropout, regularization, and data augmentation) are unnecessary.

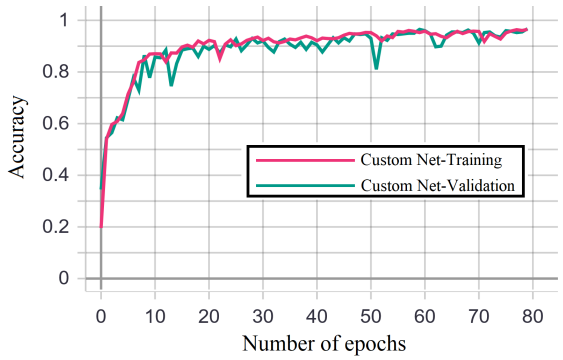


Figure 8 Training and validation accuracy versus the number of epochs for Custom Net.

For this specific dataset, the Custom Net takes 32 minutes to get trained, and on average, it takes 37.24 millisecond for Custom Net to predict the effective elastic properties of an unseen powder-bed structure. The computational time for the prediction of effective elastic properties of Custom Net is significantly shorter compared to the original DEM code (51 minutes for structure with the small number of particles and 6 hours and 34 minutes for a large number of particles).

8 Numerical examples

Several examples are presented here to verify the accuracy of the proposed CNN model for the estimation of the aggregate elastic properties of powder-beds. The described DEM model in Sections 2 to 5 is used to generate the dataset and Custom Net CNN architecture used for subsecond prediction of effective properties of the same structures. The material properties of the powder particle used to generate the dataset are listed in Table 1.

Figure 9 and Figure 10 compare the effective elastic properties evaluated by the DEM method with their corresponding predicted values using the proposed CNN method for 400 different powder-bed structures in the validation set. In general, the effective modulus

Table 1 Material properties of powder particles [9].

Property	Value
Particle mean radius (R_i)	12.5 μm
Density (ρ_i)	7800 Kg/m^3
Young's modulus (E_i)	210 GPa
Poisson's Ratio (ν_i)	0.28

of elasticity (E_x, E_y, E_z) gets larger as the packing density of the powder-bed structures increases; although, the expansion of the packing density would not necessarily cause an increase in Poisson's ratio (ν_{yz}, ν_{xz}, ν_{xy}). It is worth noting that the cubic fit to DEM evaluated, and CNN predicted values shown in Figure 9, and Figure 10 are not reflecting any fixed set of mathematical relationships. Therefore, the fast estimation of the effective elastic property of an unseen powder-bed structure could only be done with the help of forward propagation through the already trained CNN model.

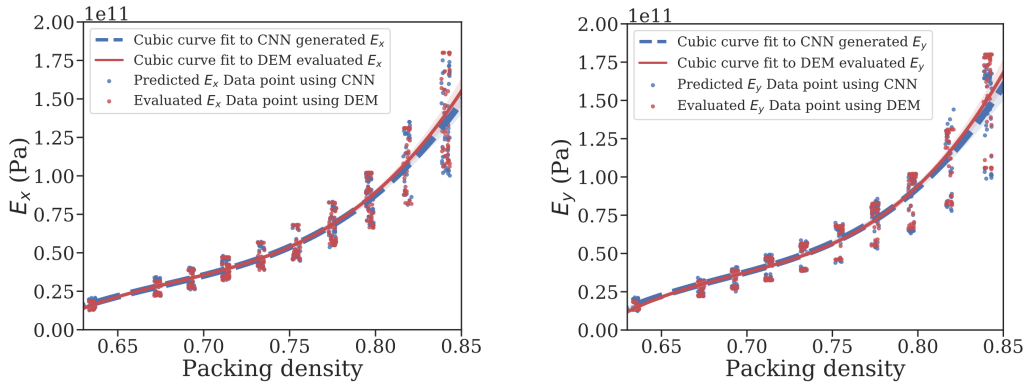


Figure 9 Scatterplot of DEM evaluated, and CNN predicted E_x (left) and E_y (right) versus Packing density validation set. The cubic fit to DEM evaluated, and CNN predicted values is also included.

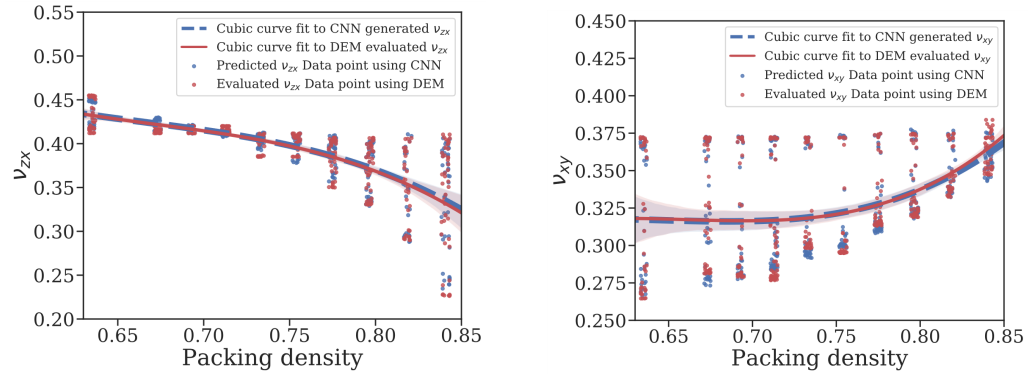


Figure 10 Scatterplot of DEM evaluated, and CNN predicted ν_{zx} (left) and ν_{xy} (right) versus Packing density validation set. The cubic fit to DEM evaluated, and CNN predicted values is also included.

Based on the results shown in Figure 9 and Figure 10, the CNN model has higher accuracy for powder-bed packing structures with lower packing density.

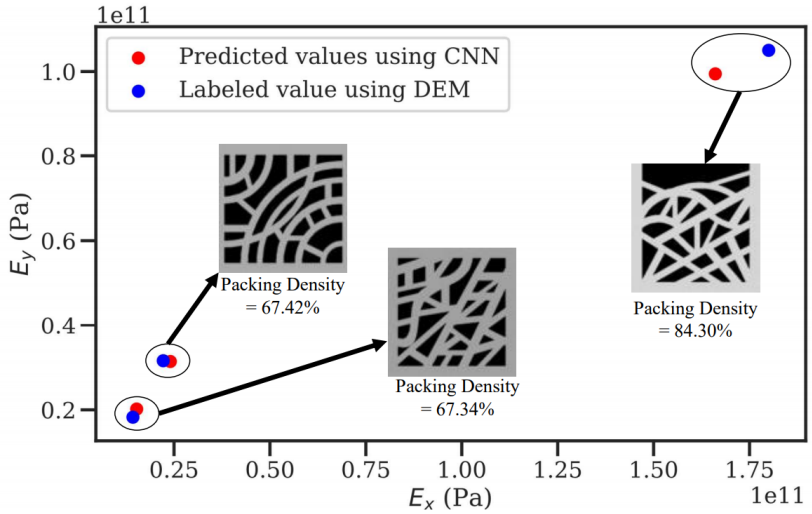


Figure 11 Scatterplot of DEM evaluated, and CNN predicted E_x versus E_y for three different powder-bed structure.

Figure 11 and Figure 12 show a comprehensive comparison between DEM evaluated and CNN predicted values of modulus of elasticity and Poisson's ratio, respectively. In each of these two figures, three powder-bed packing structure randomly selected form validation set and a scatter diagram has been used to compare two different elastic properties (E_x, E_y in Figure 11 and ν_{xz}, ν_{yz} in Figure 12). Again one can see that the CNN model is more accurate for a powder-bed packing structure that has a smaller packing density.

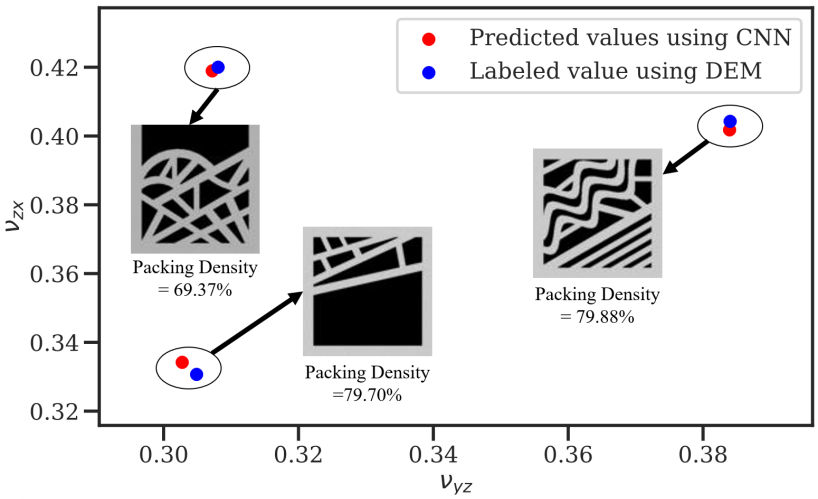


Figure 12 Scatterplot of DEM evaluated, and CNN predicted ν_{zx} versus ν_{yz} for three different powder-bed structure.

9 Conclusion and Future Work

In this work, a deep learning approach in the determination of aggregate elastic properties of powder-beds in AM structures is presented. The proposed approach has been verified by comparison with the results of DEM simulation, and for unseen powder-bed structures, the average difference in predicted elastic properties between the proposed CNN model and DEM simulation is less than 4%. The developed CNN model is much more efficient than DEM simulations because of the ability to predict the effective elastic properties without the construction of the large stiffness matrix. The computational time of CNN model is 0.0005% the time needed for solving the problem using DEM simulation. The developed CNN model will be very useful in the development of a real-time control system for predicting the aggregate elastic properties of powder-beds during the manufacturing process. The proposed CNN model is trained only for powder particles with properties listed in Table 1. In principle, a similar CNN model could be used for powder particles with different material properties by concatenating the material properties to flattened layers of CNN architecture as new neurons. Subsequently, the slightly modified CNN model must be trained using examples of powder-beds with different properties of powder particles.

References

1 K. Bagi. Microstructural Stress Tensor of Granular Assemblies With Volume Forces. *Journal of Applied Mechanics*, 66(4):934–936, December 1999. doi:10.1115/1.2791800.

2 Katalin Bagi. Stress and strain in granular assemblies. *Mechanics of Materials*, 22(3):165–177, March 1996. doi:10.1016/0167-6636(95)00044-5.

3 H. J. H. Brouwers. Particle-size distribution and packing fraction of geometric random packings. *Physical Review E*, 74(3):031309, September 2006. doi:10.1103/PhysRevE.74.031309.

4 F. Calignano, D. Manfredi, E. P. Ambrosio, L. Iuliano, and P. Fino. Influence of process parameters on surface roughness of aluminum parts produced by DMLS. *The International Journal of Advanced Manufacturing Technology*, 67(9):2743–2751, 2013. doi:10.1007/s00170-012-4688-9.

5 G. Casalino, S. L. Campanelli, N. Contuzzi, and A. D. Ludovico. Experimental investigation and statistical optimisation of the selective laser melting process of a maraging steel. *Optics & Laser Technology*, 65:151–158, January 2015. doi:10.1016/j.optlastec.2014.07.021.

6 Wei Gao, Yuanqiang Tan, and Mengyan Zang. A cubic arranged spherical discrete element model. *International Journal of Computational Methods*, 11(05):1350102, 2014. doi:10.1142/S0219876213501028.

7 Arash Gobal. *An Adaptive Discrete Element Method for Physical Modeling of the Selective Laser Sintering Process*. PhD thesis, UC Davis, 2017.

8 Arash Gobal and Bahram Ravani. An Adaptive Discrete Element Method for Physical Modeling of the Selective Laser Sintering Process, 2017. doi:10.4028/www.scientific.net/AMM.869.69.

9 Arash Gobal and Bahram Ravani. Physical Modeling for Selective Laser Sintering Process. *Journal of Computing and Information Science in Engineering*, 17(2), June 2017. doi:10.1115/1.4034473.

10 Dongdong Gu and Beibei He. Finite element simulation and experimental investigation of residual stresses in selective laser melted Ti–Ni shape memory alloy. *Computational Materials Science*, 117:221–232, 2016. doi:10.1016/j.commatsci.2016.01.044.

11 S. Haeri, Y. Wang, O. Ghita, and J. Sun. Discrete element simulation and experimental study of powder spreading process in additive manufacturing. *Powder Technology*, 306:45–54, January 2017. doi:10.1016/j.powtec.2016.11.002.

12 D. He, N. N. Ekere, and L. Cai. Computer simulation of random packing of unequal particles. *Physical Review E*, 60(6):7098–7104, December 1999. doi:10.1103/PhysRevE.60.7098.

13 George Jefferson, George K. Haritos, and Robert M. McMeeking. The elastic response of a cohesive aggregate—a discrete element model with coupled particle interaction. *Journal of the Mechanics and Physics of Solids*, 50(12):2539–2575, December 2002. doi:10.1016/S0022-5096(02)00051-0.

14 I. Kouretas and V. Paliouras. Simplified Hardware Implementation of the Softmax Activation Function. In *2019 8th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pages 1–4, May 2019. doi:10.1109/MOCAST.2019.8741677.

15 Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. URL: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

16 Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4):541–551, December 1989. doi:10.1162/neco.1989.1.4.541.

17 Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. Conference Name: Proceedings of the IEEE. doi:10.1109/5.726791.

18 Yann LeCun, Patrick Haffner, Léon Bottou, and Yoshua Bengio. Object Recognition with Gradient-Based Learning. In David A. Forsyth, Joseph L. Mundy, Vito di Gesú, and Roberto Cipolla, editors, *Shape, Contour and Grouping in Computer Vision*, Lecture Notes in Computer Science, pages 319–345. Springer, Berlin, Heidelberg, 1999. doi:10.1007/3-540-46805-6_19.

19 Hyungtae Lee and Heesung Kwon. Going Deeper With Contextual CNN for Hyperspectral Image Classification. *IEEE Transactions on Image Processing*, 26(10):4843–4855, October 2017. Conference Name: IEEE Transactions on Image Processing. doi:10.1109/TIP.2017.2725580.

20 Yousub Lee. *Simulation of Laser Additive Manufacturing and its Applications*. PhD thesis, The Ohio State University, 2015. URL: https://etd.ohiolink.edu/pg_10?0::NO:10:P10_ACCESSION_NUM:osu1440360229.

21 Xiaoxing Liu, Christophe L. Martin, Gérard Delette, and Didier Bouvard. Elasticity and strength of partially sintered ceramics. *Journal of the Mechanics and Physics of Solids*, 58(6):829–842, June 2010. doi:10.1016/j.jmps.2010.04.007.

22 Xingchen Liu and Vadim Shapiro. Homogenization of material properties in additively manufactured structures. *Computer-Aided Design*, 78:71–82, September 2016. doi:10.1016/j.cad.2016.05.017.

23 G. Miranda, S. Faria, F. Bartolomeu, E. Pinto, S. Madeira, A. Mateus, P. Carreira, N. Alves, F. S. Silva, and O. Carvalho. Predictive models for physical and mechanical properties of 316L stainless steel produced by selective laser melting. *Materials Science and Engineering: A*, 657:43–56, March 2016. doi:10.1016/j.msea.2016.01.028.

24 Mojtaba Mozaffar, Arindam Paul, Reda Al-Bahrani, Sarah Wolff, Alok Choudhary, Ankit Agrawal, Kornel Ehmann, and Jian Cao. Data-driven prediction of the high-dimensional thermal history in directed energy deposition processes via recurrent neural networks. *Manufacturing Letters*, 18:35–39, 2018. doi:10.1016/j.mfglet.2018.10.002.

25 Zhenxing Niu, Mo Zhou, Le Wang, Xinbo Gao, and Gang Hua. Ordinal Regression With Multiple Output CNN for Age Estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4920–4928, 2016. URL: http://openaccess.thecvf.com/content_cvpr_2016/html/Niu_Ordinal_Regression_With_CVPR_2016_paper.html.

26 Eric J. R. Parteli and Thorsten Pöschel. Particle-based simulation of powder application in additive manufacturing. *Powder Technology*, 288:96–102, January 2016. doi:10.1016/j.powtec.2015.10.035.

27 Arindam Paul, Mojtaba Mozaffar, Zijiang Yang, Wei-keng Liao, Alok Choudhary, Jian Cao, and Ankit Agrawal. A real-time iterative machine learning approach for temperature profile prediction in additive manufacturing processes. *arXiv:1907.12953 [cs, stat]*, August 2019. arXiv: 1907.12953. URL: http://arxiv.org/abs/1907.12953.

28 Xinbo Qi, Guofeng Chen, Yong Li, Xuan Cheng, and Changpeng Li. Applying Neural-Network-Based Machine Learning to Additive Manufacturing: Current Applications, Challenges, and Future Perspectives. *Engineering*, 5(4):721–729, August 2019. doi:10.1016/j.eng.2019.04.012.

29 Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 525–542, Cham, 2016. Springer International Publishing. doi:10.1007/978-3-319-46493-0_32.

30 Noriko Read, Wei Wang, Khamis Essa, and Moataz M. Attallah. Selective laser melting of AlSi10Mg alloy: Process optimisation and mechanical properties development. *Materials & Design (1980-2015)*, 65:417–424, January 2015. doi:10.1016/j.matdes.2014.09.044.

31 Luke Scime and Jack Beuth. A multi-scale convolutional neural network for autonomous anomaly detection and classification in a laser powder bed fusion additive manufacturing process. *Additive Manufacturing*, 24:273–286, December 2018. doi:10.1016/j.addma.2018.09.034.

32 Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*, 2015. arXiv: 1409.1556. URL: http://arxiv.org/abs/1409.1556.

33 A.K. Singh and Prakash Regalla Srinivasa. Response surface-based simulation modeling for selective laser sintering process. *Rapid Prototyping Journal*, 16(6):441–449, 2010. doi:10.1108/13552541011083362.

34 Rajwinder Singh, Rahul Rana, and Sunil Kr Singh. Performance Evaluation of VGG models in Detection of Wheat Rust, 2018. Library Catalog: www.semanticscholar.org. URL: /paper/Performance-Evaluation-of-VGG-models-in-Detection-Singh-Rana/01c8aa86f8d424f52b92efd1870caa5061bf7d64.

35 Bo Song, Shujuan Dong, Baicheng Zhang, Hanlin Liao, and Christian Coddet. Effects of processing parameters on microstructure and mechanical property of selective laser melted Ti6Al4V. *Materials & Design*, 35:120–125, March 2012. doi:10.1016/j.matdes.2011.09.051.

36 Ivar Stakgold and Michael J. Holst. *Green's Functions and Boundary Value Problems*. John Wiley & Sons, March 2011. Google-Books-ID: 8OeoISCW6qUC.

37 John C. Steuben, Athanasios P. Iliopoulos, and John G. Michopoulos. On Multiphysics Discrete Element Modeling of Powder-Based Additive Manufacturing Processes. In *Proceedings of the ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Volume 1A: 36th Computers and Information in Engineering Conference*. American Society of Mechanical Engineers Digital Collection, 2016. doi:10.1115/DETC2016-59634.

38 Tian Wang, Yang Chen, Meina Qiao, and Hichem Snoussi. A fast and robust convolutional neural network-based defect detection model in product quality control. *The International Journal of Advanced Manufacturing Technology*, 94(9-12):3465–3471, February 2018. doi:10.1007/s00170-017-0882-0.

39 Daniel Weimer, Bernd Scholz-Reiter, and Moshe Shpitalni. Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. *CIRP Annals*, 65(1):417–420, January 2016. doi:10.1016/j.cirp.2016.04.072.

40 Yangzhan Yang, Madie Allen, Tyler London, and Victor Oancea. Residual Strain Predictions for a Powder Bed Fusion Inconel 625 Single Cantilever Part. *Integrating Materials and Manufacturing Innovation*, 8(3):294–304, 2019. doi:10.1007/s40192-019-00144-5.

41 Bodi Yuan, Brian Giera, Gabe Guss, Ibo Matthews, and Sara Mcmains. Semi-Supervised Convolutional Neural Networks for In-Situ Video Monitoring of Selective Laser Melting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 744–753, January 2019. ISSN: 1550-5790. doi:10.1109/WACV.2019.00084.

42 Chen-Lin Zhang, Jian-Hao Luo, Xiu-Shen Wei, and Jianxin Wu. In Defense of Fully Connected Layers in Visual Representation Transfer. In Bing Zeng, Qingming Huang, Abdulmotaleb El Saddik, Hongliang Li, Shuqiang Jiang, and Xiaopeng Fan, editors, *Advances in Multimedia Information Processing – PCM 2017*, Lecture Notes in Computer Science, pages 807–817, Cham, 2018. Springer International Publishing. doi:10.1007/978-3-319-77383-4_79.

43 Z. Zhang, Z. J. Tan, X. X. Yao, C. P. Hu, P. Ge, Z. Y. Wan, J. Y. Li, and Q. Wu. Numerical methods for microstructural evolutions in laser additive manufacturing. *Computers & Mathematics with Applications*, 78(7):2296–2307, October 2019. doi:10.1016/j.camwa.2018.07.011.

44 Anwen Zhu, Xiaohui Li, Zhiyong Mo, and Ruaren Wu. Wind power prediction based on a convolutional neural network. In *2017 International Conference on Circuits, Devices and Systems (ICCDS)*, pages 131–135, September 2017. doi:10.1109/ICCDS.2017.8120465.

A Phase Field Modeling Approach of Crack Growth in Materials with Anisotropic Fracture Toughness

Christoph Schreiber[1]
Applied Mechanics, Technische Universität Kaiserslautern, Germany

Tim Ettrich
Technische Universität Kaiserslautern, Germany

Charlotte Kuhn
Faculty 7, Universität Stuttgart, Germany

Ralf Müller
Applied Mechanics, Technische Universität Kaiserslautern, Germany

Abstract

Within this contribution, we present a diffuse interface approach for the simulation of crack nucleation and growth in materials, which incorporates an orientation dependency of the fracture toughness. After outlining the basic motivation for the model from an engineering standpoint, the phase field paradigm for fracture is introduced. Further, a specific phase field model for brittle fracture is reviewed, where we focus on the meaning of the auxiliary parameter differentiating between material phases and the coupling of such a parameter to continuum equations in order to obtain the characteristic self organizing model properties. This specific model, as will be explained, provides the phenomenological and methodical basis for the presented enhancement. The formulation of an appropriate evolution equation in terms of a Ginzburg-Landau type equation will be highlighted and several comments on sharp interface models will be made to present a brief comparison. Following up on the basics we then introduce the formulation of a modified version of the model, which additionally to the handling of cracks in linear elastic materials under quasi static loading is also capable of taking into account the effect of resistance variation with respect to the potential crack extension direction. The strong and also the weak forms of the respective governing equations corresponding to the developed anisotropic phase field model are presented. Utilizing the weak formulation as starting point for the discretization of the two fields (displacement field and the phase field), the computational framework in terms of finite elements is introduced. We finally explain several test cases investigated within simulations and discuss the corresponding numerical results. Besides examples, which are set up to illustrate the general model properties, a comparison with crack paths obtained by experimental investigations will be presented in order to show the potential of the developed phase field model.

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases Phase field modeling, Brittle fracture, Anisotropic fracture toughness, Finite elements

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.9

Funding Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 252408385 - IRTG 2057.

[1] corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 9; pp. 9:1–9:17
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

One of the important branches of engineering is the field of fracture mechanics. Dealing with failure of technical components and structures several aspects must be considered. Crucial in that regard is crack initiation, determination of critical loads for the onset of crack growth or also the prediction of crack paths or patterns. In order to ensure a design, that combines safety, reliability and also economical aspects, fracture mechanic principles as they are proposed in e.g. [8] or in many other textbooks are incorporated. However, these principles and derivations are often based on simplified assumptions. For complex structures fracture mechanic principles are evaluated applying computational methods as for instance finite elements. Generally, the two alternative approaches are sharp interface approaches on the one hand and diffuse interface approaches on the other hand. Sharp interface approaches either have to adopt a growing crack surface via incremental remeshing algorithms (see e.g. [10]) or are based on so-called enrichment functions, which then often cause conditioning problems, in order to model propagating cracks (see e.g. [24]). The latter is referred to as extended or generalized finite element method. In contrast, within diffuse interface approaches cracks are represented by a smeared approximation. This is realized by a continuous transition of an auxiliary parameter. In terms of phase field modeling this parameter is referred to as order parameter, whose governing equation is coupled to the mechanical problem. As phase field models for solidification simulations [14, 34] propose, the evolution of the order parameter is generally driven by the criterion of energy minimization. First phase field models for fracture as e.g. [15] or [23] were initially concerned with brittle fracture under quasi static monotonic loading. The total energy considered in these models accounts for the bulk energy and also for crack energy in the sense of Griffith's theory [7], which certainly represents a keystone of fracture mechanics. Besides the evolution governed by an energy criterion another pivotal benefit of these models is that no remeshing or refining algorithms are necessary as the crack field is incorporated by an additional degree of freedom. It evolves within the preexisting domain. Accordingly, the method was extended to the fields of cohesive fracture proposed in e.g. [16] and [33], ductile fracture proposed in e.g. [17] [22] and also models taking into account inertia for dynamic fracture [28]. Also an extension to fatigue crack initiation and growth was modeled by means of the phase field approach in [30, 29, 1, 31].

Another very important class of fracture problems, which will be treated within the present work, is concerned with materials that reveal a certain anisotropy of the resistance against fracture. In contrast to an elastic anisotropy of the material, which is mainly responsible for additional parameters in the stiffness tensor and accordingly effects the mechanical response to a certain load, the directionality of the fracture resistance has a crucial impact on the path a crack will follow through the material. This effect can be detected in experimental studies as performed in e.g. [11]. To incorporate this kind of anisotropic behavior within a phase field fracture model, a critical energy release rate depending on the crack deflection direction may be introduced. This approach, which generally resembles the Maximum Energy Release Rate (MERR) criterion [35], was introduced in terms of phase field modeling by Hakim and Karma [9] and also by Li et al. [19]. In the present work we study an alternative approach to modify the crack energy, that overcomes the issue of determining the crack tip orientation and therefore provides an appropriate framework for implicit monolithic solution schemes. In Section 2 we give a brief overview in terms of directional fracture resistance and illustrate potential causes. Subsequently the complete model is derived, where we start with the basic model proposed by Kuhn and Müller [15] and continue with the enhancement accounting for the anisotropy. Section 3 deals with the implementation of the model in a finite element framework, which was utilized for the computation of the examples presented in Section 4.

2 Phase field model for anisotropic fracture

2.1 Crack deflection in materials with directionality of the fracture resistance

One of the most important principles of todays understanding of fracture phenomena goes back to the early 20th century when Alan Arnold Griffith published his theory [7], which associates the change of surface energy with crack length. For an a priori known crack in a homogeneous, linear elastic and isotropic medium under quasi static loading he proposed the energy criterion

$$\mathcal{G}_c = \frac{\mathrm{d}\Gamma}{\mathrm{d}a} = -\frac{\mathrm{d}\Pi}{\mathrm{d}a} \tag{1}$$

for onset of crack extension. In Equation (1) Π is the potential energy stored in the structure under consideration and $\mathcal{G}_c$ is the materials critical surface energy density. The infinitesimal surface energy $\mathrm{d}\Gamma = \mathcal{G}_c \mathrm{d}a$ is proportional to the crack surface and accordingly Griffith's criterion states that crack growth occurs once the increase of surface energy balances the release of potential energy. Again, this criterion assumes a crack path, that is known a priori. However, different methods were proposed to determine the deflection direction for arbitrary plane loading, as for instance the principle of local symmetry [6] or the MERR criterion [35]. The latter states that assuming the energy release rate $\frac{\mathrm{d}\Gamma}{\mathrm{d}a}$ as a function of the potential direction of propagation, the crack will chose the direction maximizing the energy release. However, if the critical energy release rate $\mathcal{G}_c$ is also a function of the virtual deflection angle the MERR criterion is no more sufficient (see [20]). This special form of an anisotropy can have different sources. A typical example are thin walled structures manufactured by (hot-)rolling or extruding processes, as illustrated in Figure 1. These processes introduce a certain deformation of the microstructure of the metal products, which leads to an elongation of grains in a specific direction, as schematically shown by the microstructures on the bottom of Figure 1. The indicated crack paths illustrate transcrystalline crack growth for the left picture and intercrystalline crack growth for the right picture, respectively. Obviously, for an equal crack length a crack propagating through the material in vertical direction has to break through a higher number of grain boundaries than a crack propagating in horizontal direction. Analogous, in terms of intercrystalline crack extension the vertical crack has a much more tortuous path over an equal length than the horizontal crack. Thus, to obtain a crack growing purely vertical, a higher force magnitude is expected than for a crack growing purely horizontally. In other terms, the resistance against cracking, which is referred to as fracture toughness, varies with the virtual crack direction. Accordingly, not only the loading quantity $\frac{\mathrm{d}\Pi}{\mathrm{d}a}$ depends on the direction but also the material parameter $\mathcal{G}_c$. An approach [13] to include such an anisotropy in the determination of the crack growth direction suggests to consider the ratio of these quantities and specify the maximum of this function of the potential crack extension angle. An elliptical interpolation is used for the quantity $\mathcal{G}_c$ with respect to the virtual cracking direction.

2.2 Phase field model for materials with directional fracture resistance

Within a large variety of physical processes different microstructures or phase compositions are generated. For example the kinetics of martensite to austenite transformation in a certain alloy. The process of solidification in metal casting as well as the cracking behavior can be modeled by a phase field. In fracture the phases are associated with broken and intact material. The tracking of an interface is problematic from an numerical perspective.

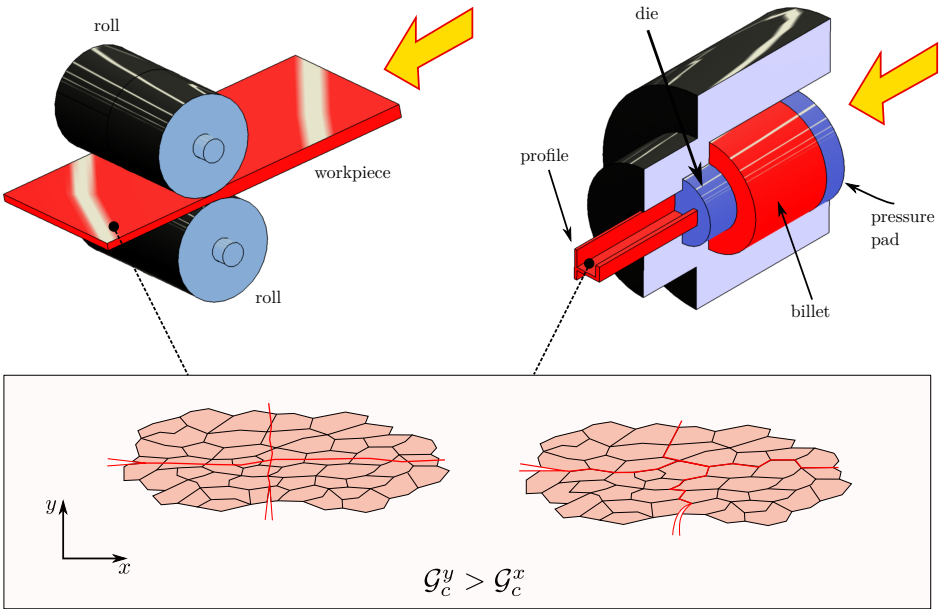


Figure 1 Illustration of two manufacturing processes (left: rolling, right: extruding) for thin walled structures and schematic illustration of microstructure with two crack paths indicated (left: transcrystalline crack propagation, right: intercrystalline crack propagation).

Especially, sharp interface models frequently encounter problems (see. e.g. [26, 25]). In terms of fracture, the nature of this issue often lies in the discretization, which is explicitly mapped to the crack surface. As an alternative, the framework of phase field modeling provides a method overcoming a number of issues of sharp interface models. Within this method an auxiliary parameter, commonly referred to as order parameter or phase field parameter, is introduced to indicated the phase at a certain location. Within the following fracture model the order parameter is $s(\mathbf{x}, t)$ and it takes the value $s = 1$ as long as the material is completely intact. Fully broken material is then indicate by $s = 0$. The transition between these values is diffuse, which conveniently overcomes the issue of discontinuities. The following sections explain in detail, how the coupling of this additional field to the deformation is done and how the evolution of the phase field $s(\mathbf{x}, t)$ is described.

2.2.1 Phase field model for brittle fracture

Within material science the evolution of the phase are commonly derived from potentials of the free energy, which mathematically are treated as functionals $F[s(\mathbf{x}, t)]$. The temporal evolution of the phase field variable, which is governed by nonlinear partial differential equations can be obtained by

$$\frac{\partial s(\mathbf{x}, t)}{\partial t} = -M \frac{\delta F}{\delta s(\mathbf{x}, t)}, \tag{2}$$

which is referred to as the time dependent Ginzburg-Landau equation and was first applied to first order phase transformations in [4]. In this equation M is a viscous regularization coefficient to control the relaxation towards stationary states. The characteristic energy

potential ψ depends on a combination of bulk energy and an interface energy such that

$$F\left[s(\mathbf{x},t)\right] = \int_{\Omega} \psi(s, \nabla s)\, \mathrm{d}V = \int_{\Omega} f(s) + \frac{1}{2}\kappa|\nabla s|^2 \, \mathrm{d}\Omega. \tag{3}$$

In this equation $f(s)$ represents the bulk energy density and a nonlocal contribution in form of a gradient term accounts for interfacial energy. The coefficient κ associated with the gradient energy term must be positive to ensure stability of one phase states. Using the free energy Equation (3) and applying the formal variation of the functional $F\left[s(\mathbf{x},t)\right]$, namely

$$\delta F\left[s(\mathbf{x},t)\right] = \int_{\Omega} \delta s \left[\frac{\partial \psi}{\partial s} - \nabla \cdot \left(\frac{\partial \psi}{\partial \nabla s}\right)\right] \, \mathrm{d}\Omega \tag{4}$$

with δs being variation of s the functional derivative can be formulated by

$$\frac{\delta F}{\delta s} = \frac{\partial \psi}{\partial s} - \nabla \cdot \left(\frac{\partial \psi}{\partial \nabla s}\right) \tag{5}$$

and Equation (2) yields

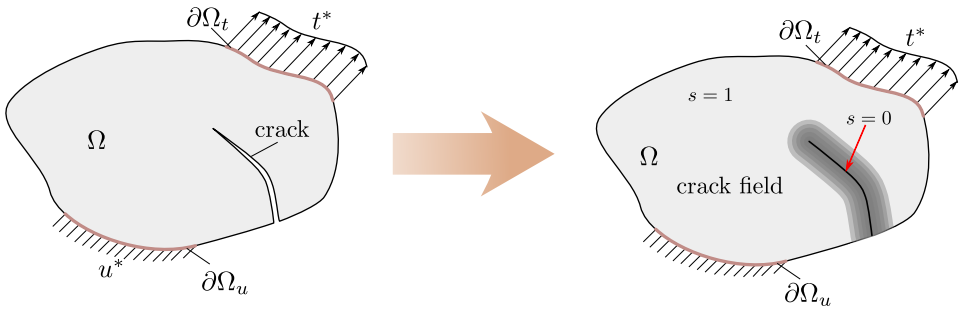
$$\frac{1}{M}\dot{s} = \nabla \cdot \left(\frac{\partial \psi}{\partial \nabla s}\right) - \frac{\partial \psi}{\partial s}. \tag{6}$$

The state $\delta F/\delta s = 0$ is a necessary condition for an extreme value of $s(\mathbf{x},t)$ and hence characterizes stationary states. Accordingly, Equation (6) can be viewed as relaxation to reach stationary states governed by the viscosity parameter $M > 0$.

In order to develop a phase field model for brittle fracture, Kuhn and Müller [15] utilized the energy potential

$$F[\boldsymbol{\varepsilon}(\mathbf{u},t), s(\mathbf{x},t)] = \int_{\Omega} \Bigg[\underbrace{(g(s)+\eta)\frac{1}{2}\boldsymbol{\varepsilon} : [\mathbb{C}\boldsymbol{\varepsilon}]}_{\psi_e} + \underbrace{\mathcal{G}_c \left(\frac{(1-s)^2}{4\epsilon} + \epsilon|\nabla s|^2 \right)}_{\psi_c} \Bigg] \mathrm{d}V, \tag{7}$$

which was proposed in [3, 2] as a regularization of the variational formulation of brittle fracture developed by Francfort and Marigo in [21]. The two contributions in Equation (7), ψ_e and ψ_c represent energies related to deformation and cracking, respectively. The first term ψ_e is the elastic energy density for small deformations with the linearized second order strain tensor $\boldsymbol{\varepsilon}$ and the isotropic fourth order stiffness tensor $\mathbb{C}$. The second term in Equation (7) may be interpreted as regularization of Griffith's surface energy. The parameter ϵ controls the width of the transition zone between the phases and can therefore be considered a regularization parameter. The multiplication of the strain energy with $g(s) + \eta$ accounts for the degradation of the stiffness in areas with $1 > s$. In this regard $g(s)$ is referred to as a degradation function modeling the material response as a variation of the phase field s. The least set of requirements for this functions is $g(s = 0) = 0$, $g(s = 1) = 1$ and $g'(s = 0) = 0$. The parameter η with $0 < \eta \ll 1$ ensures a residual stiffness $\eta\mathbb{C}$ for numerical convenience. Within Figure 2 the described approximation scheme for the fracture problem by means of the introduced phase field framework is illustrated. The crack potential ψ_c of Equation (7) consists of a nonlocal part characterized by the spatial gradient of s and a local part, characterized by the monotonous function $(1-s)^2$. This local contribution has crucial impact on the model properties (see e.g. [18]). In contrast, the double well function, which



sharp interface representation diffuse representation

Figure 2 Schematic illustration of approximation of fracture problem by means of phase field modeling: left, body with real sharp crack; right, diffuse representation of crack introducing a continuous crack field.

was applied in e.g. [12] in terms of fracture reveals a kind of naturally induced barrier between two phases. Irreversibility of the fracture process is accordingly not generally ensured by this monotonous potential. This quadratic potential, however, provides several crucial benefits, such as that the mathematical property of Γ-convergence (see. e.g. [5]) can be proven for Equation (7). Further the corresponding states of the potential for the opposite phases $s = 1$ and $s = 0$ yield the maximum distance within the relevant range $[0, 1]$ leading to a distinct separation between intact and broken material with regard to the potential energy, which prevents the crack field from growing fat. The stresses can be derived from the potential ψ by

$$\boldsymbol{\sigma} = \frac{\partial \psi}{\partial \boldsymbol{\varepsilon}} = (g(s) + \eta)\mathbb{C}\boldsymbol{\varepsilon}, \tag{8}$$

in which a hyperelastic material behavior is presumed. With Equation (7) a proper energy potential depicting a potentially fractured body is given and an evolution of $s(\mathbf{x}, t)$ with respect to time can be formulated considering the general law for phase transformation of Equation (6), which yields a Ginzburg-Landau type evolution equation

$$\dot{s} = -M\frac{\delta}{\delta s}(\psi_e + \psi_c) = -M\left\{\frac{1}{2}g'(s)\boldsymbol{\varepsilon} : [\mathbb{C}\boldsymbol{\varepsilon}] - \mathcal{G}_c\left(\frac{1-s}{2\epsilon} + 2\epsilon\nabla\cdot\nabla s\right)\right\}. \tag{9}$$

With the kinetic parameter M approaching infinity, quasi static conditions are approximated. Furthermore, this evolution equation points out the significance of the condition $g'(s = 0) = 0$ as without such a constraint the evolution of s would still continue even for broken material.

2.2.2 Incorporation of fracture toughness anisotropy

The model above predicts the crack pattern, that minimizes the total internal energy. The evolution can be interpreted as a competition between strain energy and crack energy at any time. The latter is not dependent on the potential extension direction within the original formulation [15] of brittle fracture. As stated above, a common way to incorporate directionality of the fracture resistance is to assume the critical energy release rate $\mathcal{G}_c$ as a function of the virtual crack growth direction. Declaring φ the angle between crack growth direction and a specified axis this yields $\mathcal{G}_c = \mathcal{G}_c(\varphi)$. In the context of phase field modeling this approach was studied in [9] and [19]. In this regard, the crack orientation must be

determined as the normal vector

$$\mathbf{n} = \frac{\nabla s}{|\nabla s|} \tag{10}$$

of the crack trajectory as direction for the critical energy release rate $\mathcal{G}_c(\mathbf{n})$. This definition is problematic at areas where ∇s vanishes, which renders Equation (10) impractical.
As an alternative approach $\mathcal{G}_c$ may be kept constant and instead the nonlocal part of the crack energy density $\psi_c^{\mathrm{nl}} = \mathcal{G}_c \epsilon |\nabla s|^2$ is modified in order introduce directionality. This is induced by a proper weighting of the particular components of the gradient. Visualizing the unmodified nonlocal crack potential for the 2d case in a proper axis system one obtains a paraboloidal (left illustration in Figure 3). This indicates that the direction of ∇s has no effect on the crack energy in the isotropic formulation. In contrast, if the paraboloid is deformed towards an elliptical shape, the direction of ∇s will indeed affect the total energy. For instance a gradient with a high $\frac{\partial s}{\partial x}$ contribution will shift ψ_c^{nl} to a higher level compared to a gradient with a high $\frac{\partial s}{\partial y}$ contribution with equal magnitude. Such a transformation of the nonlocal crack potential can be accomplished by substituting the identity tensor in

$$|\nabla s|^2 = \nabla s^T \mathbf{1} \nabla s \tag{11}$$

by an appropriate second order tensor $\mathbf{\Phi}$ such that the new total energy is obtained with the crack energy density taking into account the crack direction by means of the crack fields gradient with

$$\psi_c = \mathcal{G}_c \left[\frac{(1-s)^2}{4\epsilon} + \epsilon \left(\left(\frac{\partial s}{\partial x} \right)^2 \Phi_{11} + \frac{\partial s}{\partial x} \frac{\partial s}{\partial y} \Phi_{12} + \left(\frac{\partial s}{\partial y} \right)^2 \Phi_{22} \right) \right] \tag{12}$$

within a plane setting. The tensor $\mathbf{\Phi}$ may be referred to as resistance tensor and for the simplest case accounting for a modification illustrated in Figure 3 this tensor takes the diagonal form

$$\mathbf{\Phi}_0 = \begin{pmatrix} 1+\alpha & 0 \\ 0 & 1-\alpha \end{pmatrix} \tag{13}$$

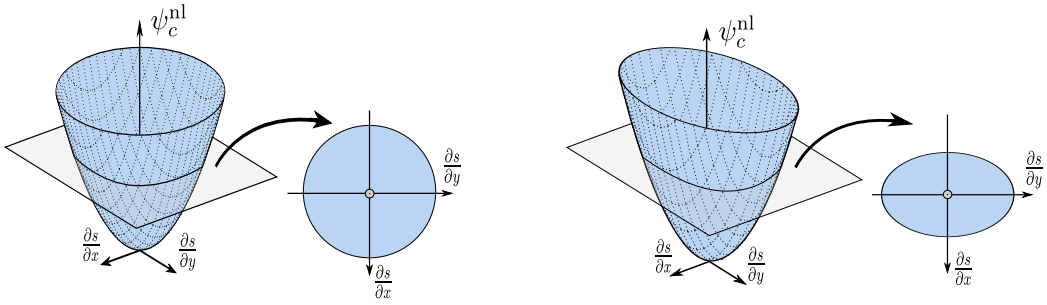
with the respective eigenvalues $1 \pm \alpha$ quantifying the degree of directionality governed by the parameter α. This formulation represents an elliptical interpolation between the values $1+\alpha$ and $1-\alpha$ with the respective reference axis represented by the eigenvectors. Suppose, the principal axes deviate from the reference system, then $\mathbf{\Phi}$ may be rotated via the rotation tensor $\mathbf{R}(\theta)$ to obtain

$$\mathbf{\Phi} = \mathbf{R}\mathbf{\Phi}_0\mathbf{R}^T = \begin{pmatrix} 1+\alpha\left(\cos^2\theta - \sin^2\theta\right) & 2\alpha \sin\theta\cos\theta \\ 2\alpha\sin\theta\cos\theta & 1-\alpha\left(\cos^2\theta - \sin^2\theta\right) \end{pmatrix}. \tag{14}$$

Incorporating this formulation of the resistance tensor in the energy functional Equation (7) yields the evolution equation of the phase field model for brittle fracture for materials with anisotropic fracture toughness

$$\dot{s} = -M \left\{ \frac{1}{2} g'(s) \boldsymbol{\varepsilon} : [\mathbb{C}\boldsymbol{\varepsilon}] - \mathcal{G}_c \left(\frac{1-s}{2\epsilon} + 2\epsilon \nabla \cdot (\mathbf{\Phi} \nabla s) \right) \right\}, \tag{15}$$

where now the orientation dependency is characterized by material parameters α and θ representing the intensity and orientation of the anisotropy, respectively. This characterization



no favored direction of gradient directionality

Figure 3 Schematic illustration of nonlocal crack potential assuming plane crack evolution of the original formulation (paraboloid in $\frac{\partial s}{\partial x}, \frac{\partial s}{\partial y}$-system) and modified formulation (elliptic paraboloid in $\frac{\partial s}{\partial x}, \frac{\partial s}{\partial y}$-system).

resembles the kind of a so-called twofold anisotropy. However implementing more general forms of the resistance tensor, the model could also account for more complex anisotropies as outlined in e.g. [19, 32].

The phase field evolution Equations (9) and (15) have the form of a inhomogeneous second order partial differential equation, which can be solved analytically only for a number of special cases. For the one dimensional case as depicted in Figure 4, Equation (15) with Equation (13) becomes

$$0 = 2(1+\alpha)\epsilon \frac{\mathrm{d}^2 s}{\mathrm{d}x^2} - \frac{s-1}{2\epsilon} \tag{16}$$

if a preexisting crack within an unloaded domain is presumed. This differential equation can be solved utilizing exponential trial functions and presuming the boundary conditions $s'(1) = s'(-1) = 0$ and $s(0) = 0$ one obtains the phase field

$$s(x) = 1 - \exp\left(\frac{-|x|}{2\epsilon\sqrt{1+\alpha}}\right). \tag{17}$$

Accordingly the shape of $s(x)$ does not solely depend on the length scale ϵ but also on the parameter α. As will be outlined in Section 4.1, the one dimensional solution can be extended in order to approximate an unidirectional crack in two dimensions. This solution is illustrated in Figure 4. The solid lines correspond to the equation of the original model [15] for isotropic materials for two different values of ϵ, where the crack becomes more narrow for a smaller ϵ. The dashed lines in contrast are obtained by Equation (17) with equal ϵ but different values for α. These plots reveal a crucial property of the modification. Assuming a basic resistance tensor of the form of Equation (13) the parameter α yields a variation of the actual length scale with the consequence of a wider crack for $\alpha > 0$ and a sharper crack for $\alpha < 0$. As the quality of the approximation of cracks by a phase field model depends on the regularization length, the intensity parameter α may therefore be chosen as little as possible relatively to ϵ. Note that the modification yields the original model for the case of zero anisotropy, namely $\alpha \to 0$.

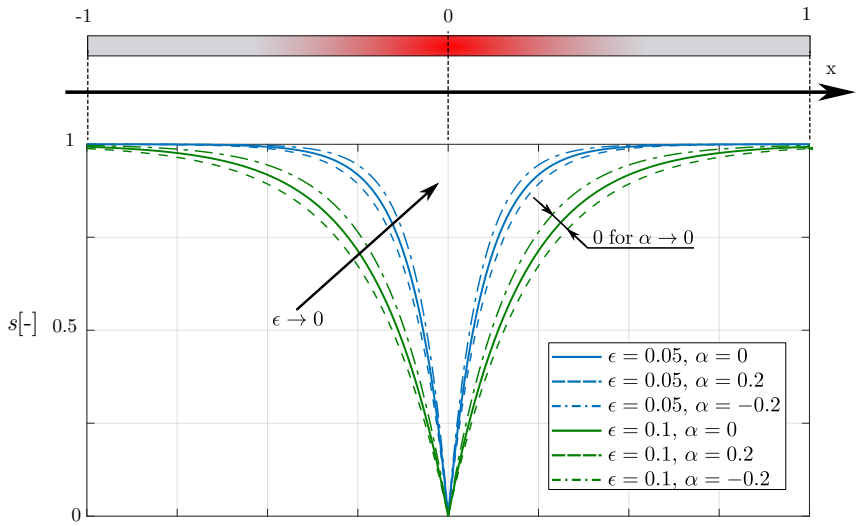


Figure 4 Illustration of the analytic solution of the proposed phase field model within one dimensional assessment of a stationary crack using different values for parameters ϵ and α.

3 Numerical implementation

The phase field model for materials with anisotropic fracture toughness derived in the previous sections is discretized and implemented into a finite element framework. With the evolution equation for the phase field Equation (15), the set of partial differential equations governing the displacements $\mathbf{u}$ and the order parameter s is

$$\begin{aligned} \mathbf{0} &= \operatorname{div} \boldsymbol{\sigma} \\ \frac{\mathcal{G}_c}{2\epsilon} &= \frac{\dot{s}}{M} + \frac{1}{2} g(s)' \boldsymbol{\varepsilon} : \mathbb{C}\boldsymbol{\varepsilon} + s\frac{\mathcal{G}_c}{2\epsilon} - \mathcal{G}_c 2\epsilon \nabla \cdot (\boldsymbol{\Phi} \nabla s)\,, \end{aligned} \tag{18}$$

neglecting volume forces in the mechanical equilibrium. The respective weak forms of these equations can be obtained by multiplying both equations with virtual quantities $\delta\mathbf{u}$ and δs, respectively. This yields

$$0 = -\int_{\Omega} \boldsymbol{\sigma} : \delta\boldsymbol{\varepsilon} \, \mathrm{d}V + \int_{\partial\Omega_t} \delta\mathbf{u} \cdot \mathbf{t}^* \, \mathrm{d}S \tag{19}$$

with vector $\mathbf{t}^*$ of prescribed tractions that act on the boundary $\partial\Omega_t$ and

$$0 = -\int_{\Omega} \left[\delta s \left(\frac{\dot{s}}{M} + \frac{1}{2} g(s)' \boldsymbol{\varepsilon} : \mathbb{C}\boldsymbol{\varepsilon} + (s-1)\frac{\mathcal{G}_c}{2\epsilon} \right) + \mathcal{G}_c 2\epsilon \nabla \delta s \cdot \boldsymbol{\Phi} \nabla s \right] \mathrm{d}V, \tag{20}$$

where integration by parts was applied. For the derivations in the following a 2d setting is presumed and Voigt's notation for symmetric tensors is applied. The field quantities $\mathbf{u}$ and s are discretized elementwise using shape functions N_I with $I = 1, \ldots, n$ and n the number of nodes per element. Accordingly,

$$\mathbf{u}(\mathbf{x}) = \sum_{I=1}^{n} N_I(\mathbf{x}) \mathbf{u}_I \qquad \text{and} \quad s(\mathbf{x}) = \sum_{I=1}^{n} N_I(\mathbf{x}) s_I, \tag{21}$$

with nodal quantities $\mathbf{u}_I$ and s_I. An analog discretization using the same shape functions N_I is applied for virtual quantities. Further, the strain tensor $\boldsymbol{\varepsilon}$ as well as the spatial gradient ∇s are discretized by

$$\varepsilon(\mathbf{x}) = \sum_{I=1}^{n} \boldsymbol{B}_I^{\mathbf{u}}(\mathbf{x})\mathbf{u}_I \qquad \text{and} \quad \nabla s(\mathbf{x}) = \sum_{I=1}^{n} \boldsymbol{B}_I^{s}(\mathbf{x})s_I, \tag{22}$$

using the respective operator matrices

$$\boldsymbol{B}_I^{\mathbf{u}}(\mathbf{x}) = \begin{bmatrix} N(\mathbf{x})_{I,x} & 0 \\ 0 & N(\mathbf{x})_{I,y} \\ N(\mathbf{x})_{I,y} & N(\mathbf{x})_{I,x} \end{bmatrix} \qquad \text{and} \quad \boldsymbol{B}_I^{s} = \begin{bmatrix} N(\mathbf{x})_{I,x} \\ N(\mathbf{x})_{I,y} \end{bmatrix}. \tag{23}$$

The vector of the internal forces, which corresponds to the displacements $\mathbf{u}$ and s at the particular node I are derived from Equations (19) and (20) as

$$\boldsymbol{F}_I^{\mathbf{u}} = \int_{\Omega} \left(\boldsymbol{B}_I^{\mathbf{u}}\right)^T \boldsymbol{\sigma} \, \mathrm{d}V \tag{24}$$

and

$$F_I^{s} = \int_{\Omega} \left[N_I \left(\frac{\dot{s}}{M} + \frac{1}{2} g(s)' \boldsymbol{\varepsilon}^T \mathbb{C} \boldsymbol{\varepsilon} + \mathcal{G}_c \frac{s-1}{2\epsilon} \right) + 2\mathcal{G}_c \epsilon \left(\boldsymbol{B}_I^{s}\right)^T \boldsymbol{\Phi} \nabla s \right] \mathrm{d}V. \tag{25}$$

The Newton-Raphson scheme is applied to solve the nonlinear system. This requires the tangential stiffness matrix associated to the framework. These particular components are obtained by derivation of the residual with respect to the unknown field quantities as

$$\boldsymbol{K}_{IJ}^{\mathbf{uu}} = \frac{\partial \boldsymbol{F}_I^{\mathbf{u}}}{\partial \mathbf{u}_J} = \int_{\Omega} \boldsymbol{B}_I^{\mathbf{u}} (g(s) + \eta) \mathbb{C} \boldsymbol{B}_J^{\mathbf{u}} \, \mathrm{d}V, \tag{26}$$

$$\boldsymbol{K}_{IJ}^{s\mathbf{u}} = \frac{\partial \boldsymbol{F}_I^{s}}{\partial \mathbf{u}_J} = \int_{\Omega} N_I g(s)' \left(\mathbb{C}\boldsymbol{\varepsilon}\right)^T \boldsymbol{B}_J^{\mathbf{u}} \, \mathrm{d}V, \tag{27}$$

$$\boldsymbol{K}_{IJ}^{\mathbf{u}s} = \frac{\partial \boldsymbol{F}_I^{\mathbf{u}}}{\partial s_J} = \int_{\Omega} \left(\boldsymbol{B}_I^{\mathbf{u}}\right)^T g(s)' \mathbb{C}\boldsymbol{\varepsilon} N_J \, \mathrm{d}V, \tag{28}$$

$$\boldsymbol{K}_{IJ}^{ss} = \frac{\partial \boldsymbol{F}_I^{s}}{\partial s_J} = \int_{\Omega} \left[N_I N_J \left(\frac{1}{2} g(s)' \boldsymbol{\varepsilon}^T \mathbb{C} \boldsymbol{\varepsilon} + \frac{G_c}{2\epsilon} \right) + 2\mathcal{G}_c \epsilon \left(\boldsymbol{B}_I^{s}\right)^T \boldsymbol{\Phi} \boldsymbol{B}_J^{s} \right] \mathrm{d}V \tag{29}$$

Finally, the damping matrix can be derived which reveals zero contributions except for the component

$$D_{IJ}^{s\dot{s}} = \frac{\partial F_I^{s}}{\partial \dot{s}} = \int_{\Omega} \frac{N_I N_J}{M} \, \mathrm{d}V. \tag{30}$$

The implicit Euler method is applied for the transient problem. To ensure a reliable convergence behavior, automatic step size adjustment is applied to ensure a stable computation even for situations of a rapid evolution of the crack field.

4 Numerical examples

The phase field model derived in the previous sections was implemented into a user routine for a quadrilateral element of the software FEAP 8.4. This routine was used for the simulations outlined in the following.

4.1 Energetic assessment of the modification

An general solution of the partial differential equation Equation (15) can not be obtained. However, for an unidirectional crack one can presume that there is no variation of the phase field with respect to the direction of crack extension and accordingly one may approximate the 2d solution by an extrusion of the one dimensional solution Equation (16). As illustrated in Figure 5a) the diffuse region at the and of the crack is neglected within this approximation. However, this does not effect the following energy assessment as it can be shown that the crack energy for this part vanishes for $\epsilon \to 0$. Considering a stationary crack in an anisotropic material as it may be characterized by Equation (13), extending solely in one coordinate direction (see Figure 5b)), the energy consumed by this crack can be approximated using Equation (17) with

$$E_c = \int_{-a/2}^{a/2} \int_{-L}^{L} \psi_c \, \mathrm{d}x\mathrm{d}y = a\mathcal{G}_c\sqrt{1+\alpha}\left[1 - \exp\left(\frac{-L}{\epsilon\sqrt{1+\alpha}}\right)\right]. \tag{31}$$

Presuming a very fine regularization ($\epsilon \to 0$) for a crack of length one this expression simplifies to

$$E_c = a\mathcal{G}_c\sqrt{1+\alpha} \qquad \text{for } \epsilon \to 0 \tag{32}$$

Note that this derivation can be repeated for a crack under the same assumptions but extending purely in orthogonal direction. The respective surface energy would then be obtained as $E_c = a\mathcal{G}_c\sqrt{1-\alpha}$.

In order to prove the accuracy, these approximations may be verified by means of respective 2d simulations with the proposed phase field model. The set up for this stationary simulation is indicated in Figure 5b). Due to the symmetry of the problem with respect

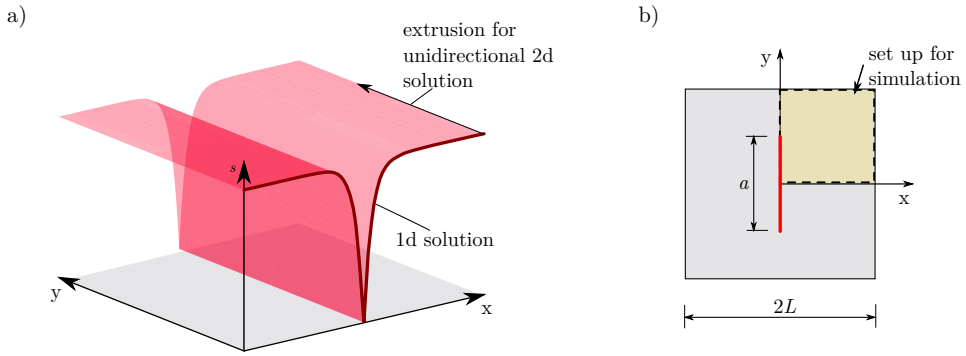


Figure 5 Schematic illustration of the extension of the one dimensional phase field solution to an unidirectional crack: a) segment of surface plot of the crack field; b) set up for stationary simulations.

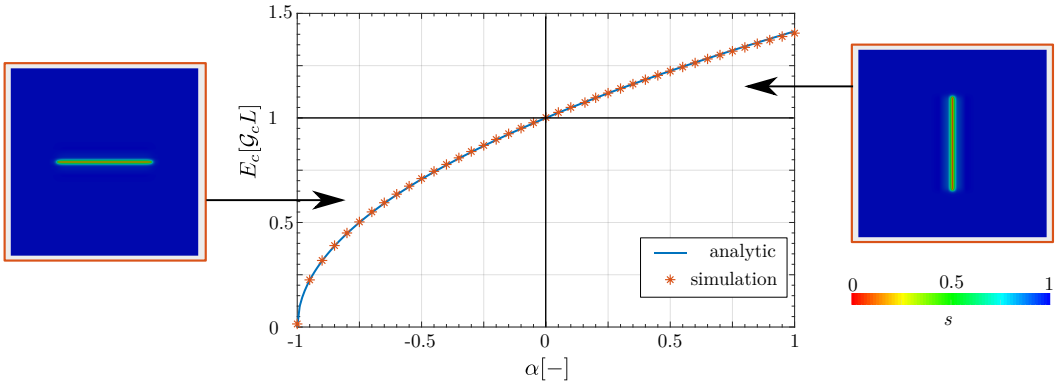


Figure 6 Comparision of analytic energy estimate of anisotropic model with results obtained from stationary finite element simulations.

to the x and y axes only a quarter of the domain was discretized using 12544 nodes. The mesh was refined in the region containing the crack in order to obtain an accurate solution of the crack field and to enable for small values of the length scale ϵ. Several simulations where run using different values of α in the range $-1 \leq \alpha \leq 1$, where simulations with $\alpha > 0$ represent cracks extending in y-direction and $\alpha < 0$ represent crack extending in x-direction. The crack energy was numerically integrated in the domain and multiplied by the factor 4. A comparison of the theoretical estimate of the total crack energy with the energy obtained by the simulations is shown in Figure 6. It shows agreement of the analytic solution with the simulation results. Therefore, it illustrates the effect of the proposed modification in terms of an energetic view.

4.2 Plate with sharp notch

The aim of this example is to illustrate the abilities of the developed model with regard to predicting crack paths. The resistance tensor $\mathbf{\Phi}$ entering the evolution equation of the phase field incorporates the two parameters α and θ. These parameters are supposed to quantify the intensity and direction of the anisotropy of the fracture resistance. In order to illustrate the sensitivity of the model to a parameter variation, a notched plate was set up. The set up is depicted in Figure 7. In all simulations, displacements were blocked in vertical direction for

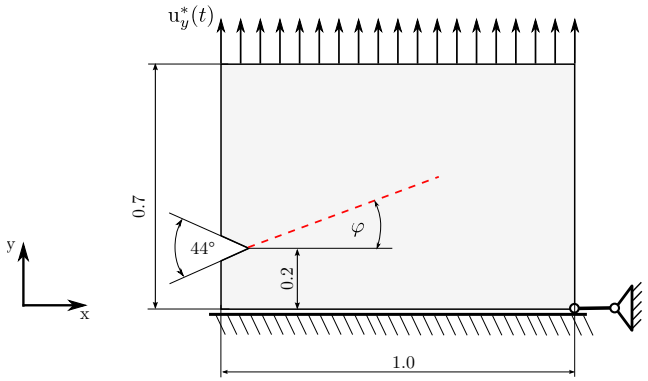


Figure 7 Illustration of set up for phase field simulation of a plate with notch.

nodes of the lower boundary edge and in horizontal direction for the node at the lower right corner of the plate. A spatially constant displacement load $u_y^*(t)$ was applied for the top edge nodes as monotonically increasing function. The specimen was discretized by means of 14899 nodes and 14406 elements with almost equal edge length. The length scale was kept constant for all simulations at a value of $\epsilon = 0.015$. For this particular set up, cracks will initiate at the notch tip. The initial deviation of the crack from a pure horizontal path, is quantified by the angle φ as indicated in Figure 7. The results for several simulations are illustrated in Figure 8. This comparison clearly indicates that both parameters have significant effect on the crack growth direction. The upper left contour plot shows a horizontal crack path, which is expected for the applied displacement load with both parameters set to zero. In contrast if α or θ are increased the crack paths reveal a deflection angle φ. For a constant orientation θ this deflection angle increases with increasing the value of intensity α.

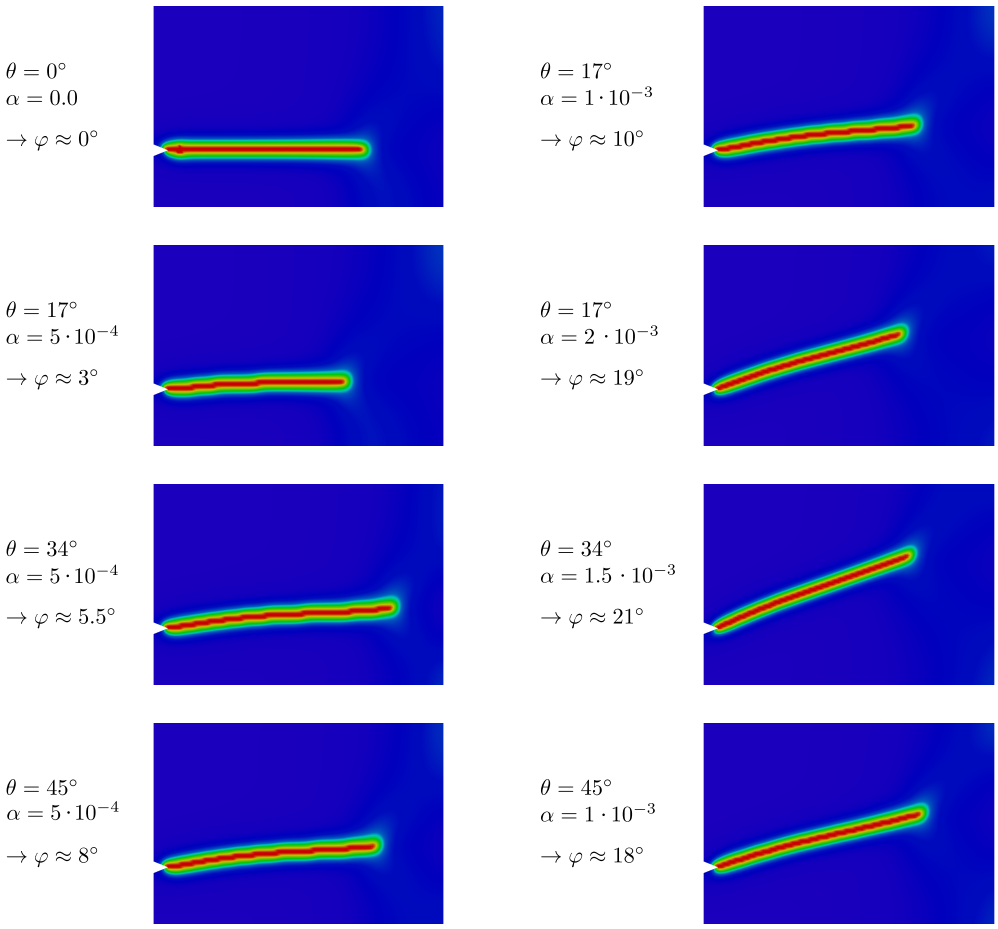


Figure 8 Comparison of deflection angle of crack paths for different parameter sets obtained from FEM simulations with the proposed anisotropic fracture model.

4.3 Plate with hole

The numerical example outlined in the following was set up in order to verify the practical usage of the modified phase field model. In the focus is an assessment of the model with respect to experimental findings from a study done by Judt et al. [11]. Plates of rolled

aluminum sheets where subjected to tensile testes. The aluminum raw material revealed a twofold anisotropy as described within the previous section. The geometry of the specimens is depicted in Figure 9a), which also illustrates the set up for the phase field simulations. The anisotropy of the plates was determined with respect to the indicated x-y-system. Accordingly, the resistance tensor reveals a structure as in Equation (13). Phase field simulations with a monotonously increasing displacement load in positive y direction where performed for different parameterizations of α. The results are illustrated in Figure 9 by the three contour

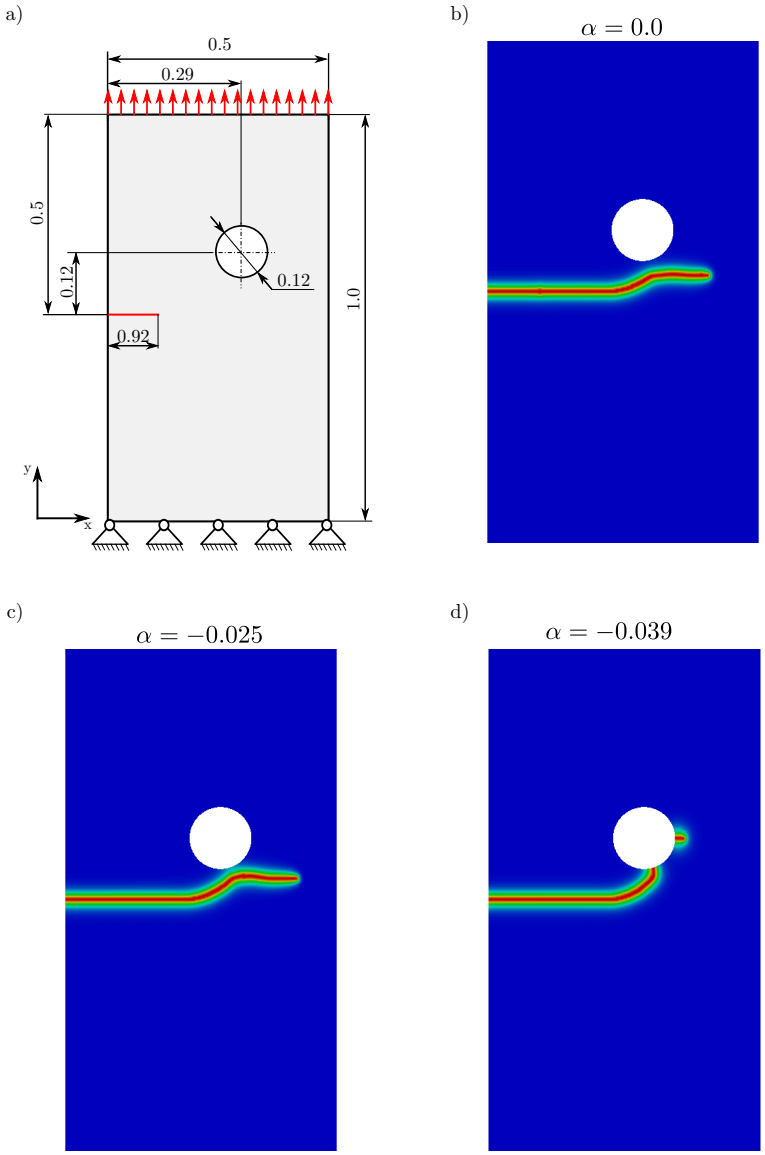


Figure 9 Set up and results (contour plots of phase field variable s) from phase field simulations with different values for α: b) $\alpha = 0.0$, c) $\alpha = -0.025$, d)$\alpha = -0.039$.

plots of the phase field variable s. The plot Figure 9d) shows the crack path obtained by the simulation with the highest magnitude for α. This crack path is quiet similar to the crack path found in [11] by experiments. As the contour plots Figure 9b) and c) show, the phase field model predicts that the induced crack passes the hole instead of growing into it. This is

due to the lower magnitude of the anisotropy, which then favors the y direction less for crack extension. Note that the crack predicted by the simulation with $\alpha = -0.025$ lies between the other two cases. Further the crack patterns shown in Figure 9b) and c) correlate with the crack paths found in [11] by a conventional crack simulation technique.

5 Concluding remarks

A phase field fracture model has been presented, which is able to account for an anisotropic fracture resistance. The anisotropy influences the path of a crack growing through the material. The nonlocal part of the regularized crack energy density is modified in order to enhance the isotropic phase field model for brittle fracture. As the evolution is governed by energy minimization, the weighting of the respective components of the spatial gradient of the order parameter includes the directionality regarding crack growth. The characteristic of the anisotropy is quantified by two additional parameters within the governing equations of the model. By means of analytic considerations it is explained how the modification effects the crack energy with respect to the parameterization. Comparing these derivations with numerical results a good agreement is confirmed. Further simulations show that cracks tend to grow in the direction favored by the parameterization. Good agreement with experimental results is also obtained for simulated crack patterns in holed plates. Accordingly, the approach we follow within this contribution seems to provide a sound basis for even more complex anisotropies. With regard to further investigations the introduced resistance tensor may be modified in order to represent materials with more complex polar plots of the critical energy release rate (see e.g. [19, 27]).

References

1 R. Alessi, S. Vidoli, and L. DeLorenzis. A phenomenological approach to fatigue with a variational phase-field model: The one-dimensional case. *Engineering Fracture Mechanics*, 190:53–73, 2017.

2 B. Bourdin. Numerical implementation of the variational formulation of brittle fracture. *Interfaces and Free Boundaries - INTERFACE FREE BOUND*, 9:411–430, 2007.

3 B. Bourdin, G. A. Francfort, and J-J. Marigio. Numerical experiments in revisited brittle fracture. *Journal of the Mechanics and Physics of Solids*, 48:797–826, 2000.

4 S. Chan. Steady-state kinetics of diffusionless first order phase transformation. *The Journal of Chemical Physics*, 67(12):5755–5762, 1977.

5 E. De Giorgi and G. Dal Maso. Γ-convergence and calculus of variations. In *Mathematical Theories of Optimization*, pages 121–143, Berlin, Heidelberg, 1983. Springer.

6 Robert V. Goldstein and Rafael L. Salganik. Brittle fracture of solids with arbitrary cracks. *International Journal of Fracture*, 10:507–523, 1974.

7 A. A. Griffith. The phenomena of rupture and flow in solids. *Philosophical Transactions of the Royal Society of London*, 221:163–198, 1921.

8 D. Gross and Th. Seelig. *Bruchmechanik–Mit einer Einfürung in dei Mikromechanik*. Springer, Heidelberg, 5 edition, 2011.

9 V. Hakim and A. Karma. Laws of crack motion and phase-field models of fracture. *Journal of the Mechanics and Physics of Solids*, 57(2):342–368, 2009.

10 P. O. Judt and A. Ricoeur. Crack growth simulation of multiple cracks systems applying remote contour interaction integrals. *Theoretical and Applied Fracture Mechanics*, 75:78–88, 2015.

11 P. O. Judt, A. Ricoeur, and G. Linek. Crack path prediction in rolled aluminum plates with fracture toughness orthotropy and experimental validation. *Engineering Fracture Mechanics*, 138:33–48, 2015.

12 A. Karma, D. Kessler, and H. Levine. Phase-field model of mode iii dynamic fracture. *Physical Review Letters*, 87:045501, 2001.

13 A.P. Kfouri. Crack extension under mixed-mode loading in an anisotropic mode-asymmetric material in respect of resistance to fracture. *Fatigue and Fracture of Engineering Materials and Structures*, 19(1):27–38, 1996.

14 Ryo Kobayashi. Modeling and numerical simulations of dendritic crystal growth. *Physica D: Nonlinear Phenomena*, 63(3):410–423, 1993.

15 C. Kuhn and R. Müller. A continuum phase field model for fracture. *Engineering Fracture Mechanics*, 77:3625–3634, 2010.

16 C. Kuhn and R. Müller. A discussion of fracture mechanisms in heterogeneous materials by means of configurational forces in a phase field fracture model. *Computer Methods in Applied Mechanics and Engineering*, 312:95–116, 2016.

17 C. Kuhn, T. Noll, and R. Müller. On phase field modeling of ductile fracture. *GAMM Mitteilungen*, 39:35–54, 2016.

18 C. Kuhn, A. Schlüter, and R. Müller. On degradation functions in phase field fracture models. *Computational Materials Science*, 108:374–384, 2015.

19 B. Li, C. Peco, D. Millán, I. Arias, and M. Arroyo. Phase-field modeling and simulation of fracture in brittle materials with strongly anisotropic surface energy. *International Journal for Numerical Methods in Engineering*, 102(3-4):711–727, 2015.

20 L. Ma and A. Korsunsky. On the use of vector j-integral in crack growth criteria for brittle solids. *International Journal of Fracture*, 133, 2005.

21 J. J. Marigo. Modelling of brittle and fatigue damage for elastic material by growth of microvoids. *Engineering Fracture Mechanics*, 21:861–874, 1985.

22 C. Miehe, F. Aldakheel, and T Stephan. Phase-field modeling of ductile fracture at finite strains. a robust variational-based numerical implementation of a gradient-extended theory by micromorphic regularization: Phase field modeling of ductile fracture. *International Journal for Numerical Methods in Engineering*, December 2016. doi:10.1002/nme.5484.

23 C. Miehe, F. Welschinger, and M. Hofacker. Thermodynamically consistent phase-field models of fracture: Variational principles and multi-field fe implementations. *International Journal for Numerical Methods in Engineering*, 83(10):1273–1311, 2010.

24 P. O'Hara, J. Hollkamp, C.A. Duarte, and T. Eason. A two-scale generalized finite element method for fatigue crack propagation simulations utilizing a fixed, coarse hexahedral mesh. *Computational Mechanics*, 57:55–74, 2016.

25 N. Provatas and K. Elder. *Phase Field Methodes in Material Science and Engineering*. Wiley, Berlin, 1 edition, 2010.

26 A.H. Rajkotwala, A. Panda, E.A.J.F. Peters, M.W. Baltussen, C.W.M. [van der Geld], J.G.M. Kuerten, and J.A.M. Kuipers. A critical comparison of smooth and sharp interface methods for phase transition. *International Journal of Multiphase Flow*, 120:103093, 2019.

27 B. Roman, E. Hamm, and F. Melo. Forbidden directions for the fracture of thin anisotropic sheets: An analogy with the Wulff plot. *Physical Review Letters*, 110, April 2013. doi:10.1103/PhysRevLett.110.144301.

28 A. Schlüter, A. Willenbücher, C. Kuhn, and R. Müller. Phase field approximation of dynamic brittle fracture. *Computational Mechanics*, 54:1141–1161, 2014.

29 C. Schreiber, C. Kuhn, and R. Müller. Phase field modeling of cyclic fatigue crack growth under mixed mode loading. *Computer Methods in Materials Science*, 19:50–56, 2019.

30 C. Schreiber, C. Kuhn, R. Müller, and T. Zohdi. A phase field modeling approach of cyclic fatigue crack growth. *International Journal of Fracture*, 2020.

31 M. Seiler, P. Hantschke, A. Brosius, and M. Kästner. A numerically efficient phase-field model for fatigue fracture – 1d analysis. *PAMM*, 18(1), 2018.

32 S. Teichtmeister, D. Kienle, F. Aldakheel, and M.A. Keip. Phase field modeling of fracture in anisotropic brittle solids. *International Journal of Non-Linear Mechanics*, pages 100–118, July 2017.

33 C. V. Verhoosel and R. de Borst. A phase-field model for cohesive fracture. *International Journal for Numerical Methods in Engineering*, 96(1):43–62, 2013.

34 A.A. Wheeler, B.T. Murray, and R.J. Schaefer. Computation of dendrites using a phase field model. *Physica D: Nonlinear Phenomena*, 66(1):243–262, 1993.

35 C. H. Wu. Maximum-energy-release-rate criterion applied to a tension-compression specimen with crack. *Journal of Elasticity*, 8:235–257, 1978.

Is Smaller Always Better? - Evaluating Video Compression Techniques for Simulation Ensembles

Patrick Ruediger
Visual Information Analysis Group, Technische Universität Kaiserslautern, Germany

Christoph Garth
Scientific Visualization Group, Technische Universität Kaiserslautern, Germany

Hans Hagen
Computergraphics and HCI Group, Technische Universität Kaiserslautern, Germany

Heike Leitte
Visual Information Analysis Group, Technische Universität Kaiserslautern, Germany

Abstract

We provide an evaluation of the applicability of video compression techniques for compressing visualization image databases that are often used for in situ visualization. Considering relevant practical implementation aspects, we identify relevant compression parameters, and evaluate video compression for several test cases, involving several data sets and visualization methods; we use three different video codecs. To quantify the benefits and drawbacks of video compression, we employ metrics for image quality, compression rate, and performance. The experiments discussed provide insight into good choices of parameter values, working well in the considered cases.

2012 ACM Subject Classification Computing methodologies → Image compression; Applied computing → Physics; Applied computing → Engineering

Keywords and phrases Image Database, CinemaDB, Video Compression, Evaluation, Benchmark, In-situ

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.10

Funding This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057.

1 Introduction

Today's supercomputer architectures allow computational scientists to perform research with increasingly complex models producing high-resolution, high-fidelity data in increasingly shorter times. It is often no longer possible to store data set at full resolution, and it is also extremely inefficient and expensive to transfer such large data sets from tertiary storage into memory for analysis. Consequently, *in situ* visualization – i.e., the generation and storage of visualizations/images done as part of an ongoing computer simulation – has been developed and as a means to avoid most of the difficulties one would have to address when performing large-scale data visualization via traditional approaches [21, 6]. Nevertheless, while an in situ approach is advantageous in many ways it limits the possibilities of "complete data exploration" to the a set of pre-generated visualizations. Ahrens et al. [1, 23] recognized that generating a large set of images covering the underlying parameter value space well still allows a scientist to meaningfully analyze and discover important model behaviors while accelerating the exploratory process significantly as all visualizations already have been generated.

For complex visualizations generated, for example, by sampling large and/or high-dimensional visualization parameter spaces, the databases needed for storage can still be of substantial size. To reduce data size and thereby make possible a more efficient management

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 10; pp. 10:1–10:18

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

and processing of visualization databases, we adapt image and video compression techniques. The goal is to leverage inter-image similarity typically encountered in video image frame sequences and used to achieve substantial compression. By "linearizing" a visualization image database it is possible to adapt video compression in support of much more efficient data analysis. This approach was first investigated by Berres et al. [7], where it was shown that approach is feasible and beneficial. However, their linearization approach and choice of compression parameter values, e.g., video codec and image encoding, were ad hoc. Our work is motivated by the goal of gaining more insight into the effects of these parameters concerning overall usefulness of video-compressed image databases.

We present the results of a broader investigation of the different aspects of applying video compression to visualization image databases. Specifically, we consider the following issues:

- We quantitatively evaluate video compression for four different data sets with significantly different image characteristics that are expected to affect compression. We specifically consider streamline and isosurface visualizations. We employ three commodity, general-purpose and easy-to-use video codecs (H.264 [26], H.265 [11], VP9 [15]). We consider several quality metrics and compression/de-compression efficiency in relationship to compression ratio, serving as a central parameter for all three codecs.
- Extending the research of Berres et al. [7], who considered the compression of RGB images, we take into account implementation issues of visualization image databases. Specifically, we compress scalar value and depth images; perform composition of visualization images; empirically demonstrate that video compression can be applied successfully to visualization images.
- We investigate the effects of linearization of a high-dimensional visualization image space on compression efficiency and show that ordering matters and a "good order" can significantly improve the final compression rate.

Based on our prior work [4], we suggest parameters suitable for general-purpose use.

After discussing related work in Section 2, we describe the used experimental setup, i.e., the compression pipeline (Section 3) and its major parameters, followed by a description of the metrics used for evaluation 4. Quantitative results are provided and analyzed in Section 5, and we discuss practical recommendations. Finally, we address limitations of our study and provide conclusions in Section 6.

2 Related Work

There is a rich body of work focusing on data reduction for computational model output. Among the typical approaches are multi-resolution techniques, adaptive refinement, compression; a survey and overview of techniques typically utilized in situ is given by Li et al. [18].

Common to all these techniques is that only reduced data is available for post hoc analysis and visualization. Often, explicit error bounds are not available, or very difficult to estimate except in specific circumstances (e.g. [16]). Thus, accuracy of analysis is in direct competition with data reduction, where it is essential to hit the sweet spot between reduction rate and data quality [3].

Taking an alternate approach, in situ visualization [21, 6] generates visualization imagery on full-fidelity data while it resides in memory, both eschewing storage costs as well as producing accurate images, but in turn limiting the flexibility of exploration. The Cinema approach [1] generates a browsable, parameter-dependent database of visualization images that facilitates post hoc exploration in a manner that is satisfactory for many applications (e.g., [23]). Through clever implementation, such as storing scalar value images and depth

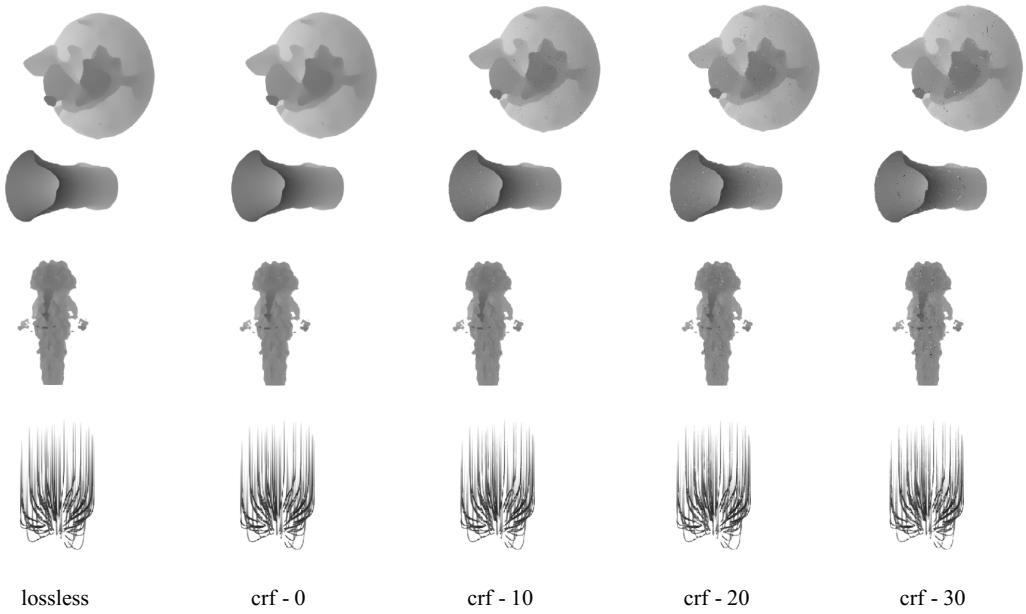


Figure 1 Depth Images before and after compression with the H.264 codec. The visually detectable differences are very small. Starting at crf= 10 the noise in the images is increasing with each step. Still most of the topological structures remain intact.

images instead of RGB images, compositing and color mapping can be employed to keep the image database size small. Lukasczyk et al. showed that under certain conditions it is possible to reconstruct parameter sets not stored in the database, such as e.g. camera positions, further enabling free exploration [20]. However, for very large parameter spaces that arise when combining many different visualizations in the interest of exploration, the visualization image database can become very large, again making its storage and use difficult. For specific scenarios, optimized image formats can be defined, e.g. in the case of volume rendering [28] or contour tree analysis [8]. However, this only marginally reduces the size of corresponding image databases. Image database storage also typically utilizes image compression techniques such as e.g. wavelet compression [30] or commodity image compression codecs (e.g. JPEG) in both lossy and lossless modes. However, compressing each image individually cannot leverage the high degree of similarity between images corresponding to closely neighboring parameter settings.

Concerning the delivery of in situ-generated visualization images, Ellsworth et.al.[12], Kageyema et.al.[17], and Biedert et al. [9] demonstrated the effectiveness of video compression to stream high-fidelity images rendered on a supercomputer to a desktop client. Applying this insight to visualization image databases, Berres et.al. [7] showed that visualization image databases are suitable for compression with video codecs, observing compression rates between 14% and 25%. Earlier, Sohn et.al.[25] proposed the idea of a specifically designed video compressor for volumetric images, using wavelet transformation and a temporal encoding to make efficient use of empty spaces. While they are maintaining good retrieval performances, the compression rates of 0.29% - 0.68% they achieve show their technique to be less effective.

Based on these prior works, and especially the proof-of-concept work of Berres et al. [7], we assume that video compression is a suitable way for reducing the size of large visualization image data bases. However, it is unclear which factors affect compression rate (i.e., data

reduction), image quality, and retrieval performance, which are the key pertinent aspects to consider when using compression for visualization image databases. In this paper, we investigate these aspects in a more comprehensive quantitative experiment.

When considering lossy compression, image quality is balanced against size reduction. In an excellent visual evaluation study, Turton et al. [4] showed that good compression rates can be achieved while preserving image quality, and found that the *structural similarity metric* [29] (SSIM) to be a good general indicator for image quality when comparing lossy compression to ground truth. In addition to other difference metrics such as PSRN, we hence base our evaluation heavily on the SSI metric.

3 Experimental Setup

In the following, we describe the pipeline we employ to carry out our study. A conceptual overview is given in Figure 2. To generate the visualization image database, we use Para-

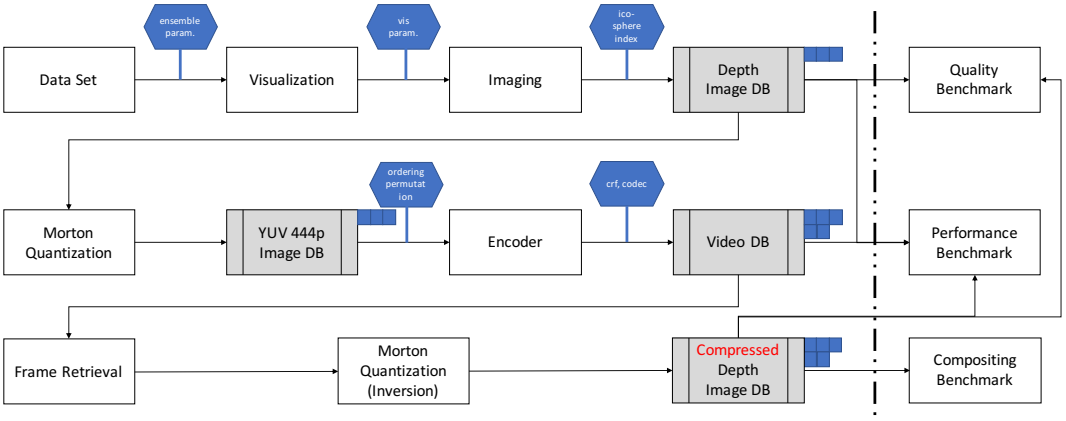


Figure 2 Overview of the image database generation, compression, and decompression pipeline underlying our study.

View [2] in combination with the TTK framework [27]. Prototype visualization pipelines are created in ParaView and stored as state files, which we then parameterize. The filters `ttkCinemaImaging` and `ttkCinemaWriter` are then used to render the images for a Cartesian product of parameter samples; including simulation time, camera positions, and isovalue (where appropriate). Camera positions are located on vertices of a spherical grid, and the cameras are aimed towards the data sets' centers.
Following the reasoning of Ahrens et al. [1], we focus on a modern implementation of visualization image databases based mainly on storing depth images instead of color images, as the former can be composited to combine different visualizations, avoiding a combinatorial size of combined visualization image databases. This allows us further to investigate the effects of compression on compositing as an additional benchmark. While we do not focus on scalar value images (used to achieve color mapping post hoc), our results (in particular, error metrics) apply directly to this class of images as well.

As an intermediate product, we obtain a depth image visualization database, which associates a depth image with each parameter set and represents the ground truth for error measurements. Image resolution is chosen as 512x512 throughout the entire study.

3.1 Data Sets and Visualizations

In the interest of covering a wide range of practical scenarios, we consider typical visualizations for four different data sets with very different characteristics.

3.1.1 Viscous Fingering

The data set is the result from a simulation of salt dissolving in water. The domain consists of a cylindrical flow, at the top of the cylinder a solid body of salt is placed that is dissolved by the water. The resulting fields are the velocity of the flow and concentration of salt in the water. The data set is time varying and consists of multiple parameter settings.

3.1.2 Geodynamo

This data set is the result from a numerical simulation of the earth's liquid outer core based on the effect of the geodynamo. The resulting domain is covering the whole outer core and is stored in a spherical unstructured grid. The resulting fields like pressure, temperature, velocity and magnetic field are characterized by high turbulence and large structures. The main challenge in this data set is its spherical domain and structures which are mostly hard to deal with standard approaches for cubic domains. In addition the time steps are in the order of thousands of years.

3.1.3 Jet Flow

The jet flow is an artificial unsteady flow simulation resembling the outlet of a jet engine. It is a well studied example data set for flow visualization and analysis and has very characteristic features. The resulting fields are the velocity and the temperature.

3.1.4 Metal Sheet Deformation

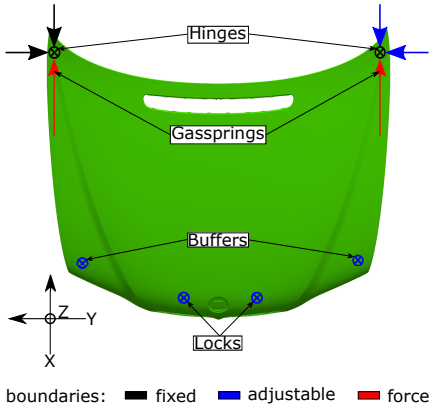


Figure 3 Sheet metal car body part used in automotive industry. The used boundaries for the simulation are high-lighted, with external forces used via gas springs. The coordinate system for this part is shown.

For a more application-driven view, a real-world example from the automotive industry was chosen. The geometry is an engine hood. This part has two hinges, two locks and two buffers as mechanical boundaries attaching the hood to the chassis, see Figure 3. A finite element (FE) simulation predicts deflections during the assembly process of the part.

However, the final shape of a material part can vary due to production uncertainties. To deal with uncertainties and tolerances during the production, the engine hood's boundaries, i.e., hinges, lock and buffers, are adjustable. Adjusting these boundaries properly to obtain an acceptable gap and flushness is a challenging task [13]. A post-assembly measurement induces necessary corrections. The goal of the proposed method is to find the best set of changes from measured deviations, which forms the optimal set of adjustments. The method uses as input an assemble of statistical distributed simulations that cover the solution space spanned by the available adjustment possibility of each boundary. The used car hood is an assembly containing seven individual sheet metal parts, connected by spot welds and different types of adhesives. Based on the CAD files, a simulation model was created by meshing the geometry with 3D-shell elements and connecting the assembly considering spot weld, adhesive positions, and thicknesses of components. The material model is linear-elastic with an e-modulus of 210Gpa and a Poisson rate of 0.3. Two fixed, external loads, modeling the gas springs near the hinges with the magnitude of 580N each, complete the model.

3.2 Image Database Linearization

We linearize the depth image database by enumerating the Cartesian product parameter space using an arbitrary ordering of axes. We hypothesize (and empirically confirm, cf. Section 5) that ordering axes in a manner that the fastest axis (along which subsequent images lie in the linearization) should be chosen such that the inter-image similarity is large to benefit compression efficiency.

Most video codecs assume that the changes from one frame to the other happens to be smoothly and in a predictable way over a longer period (see the preset parameters [10]). Therefor we assume that the best compression performance results will occur for a smooth ordering of the images. We are investigating the impact of camera path, data time step, ensemble parameter and visualization parameter as different approaches for the dominating ordering axis (see Fig.4).

Camera Location

The camera parameters are usually the first parameters that are used while creating an image data base. A collection of images ordered by camera angles is similar to tracking shots in video sequences, which are the target of video compression techniques. Based on the number of viewpoints that are needed to capture a visualized data set, however, it is not necessarily the axis with the smoothest transition between successive images. Especially if storing a minimum amount of images needed to visually adequately reconstruct geometry [20], camera location index as the fastest axis is not necessarily a good choice.

Data Time

For unsteady data sets, time provides a natural parametrization. In practice, memory or I/O constraints determine the step size between successive discrete time points. Furthermore, considering ensemble data, step sizes may vary per ensemble member. The chosen visualization techniques also plays a crucial role here, while some techniques will result on bigger visual changes with small changes in the time, others may behave in the opposite direction. For our setup, we therefore do not rely on smooth transitions between images when choosing the time as the dominant axis.

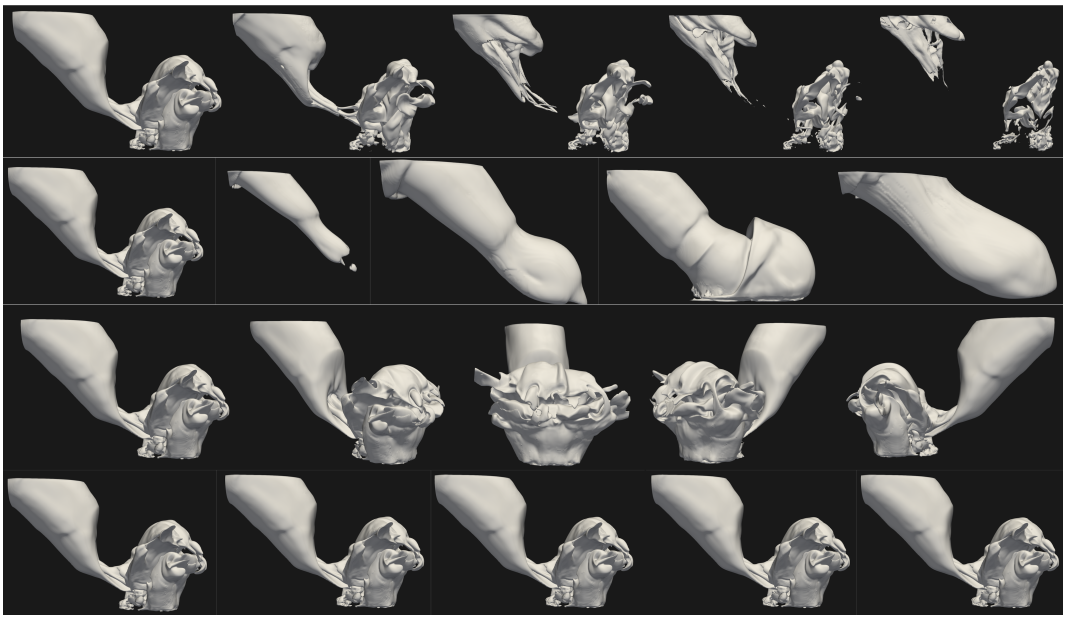


Figure 4 Showing different ordering strategies for one data set. Depending on which axis is chosen to be the dominant one, the similarity between successive images could vary dramatically. From top to bottom: Iso-value, Ensemble Parameter, Camera rotation, Simulation time step.

Ensemble parameter space

In ensemble data sets we are particularly interested in the differences that relate to changes in the input parameter space, which leads to similar effects as for setting the time step size mentioned above. For our investigation, we therefore expect that it behaves similar to the time ordering for unsteady data sets if they have a matching resolution of the step sizes.

Isovalue

Concerning algorithmic parameters that change the visualization result, we consider isovalue in the cases described above as a proxy for more general settings. Most notably, it is one of the most often changed parameters when exploring a data set. In general, variation between images should be small under a fine-grained sampling, as the surface varies slowly and smoothly with change in isovalue. Thus, we hypothesize that this parameter is a good choice for the fastest axis in the linearization if the sampling is not too coarse.

3.3 Depth Image Encoding

Before applying video compression to depth images, these must be transferred into a format that is suitable for ingestion by video codecs. While depth images encode a single scalar in the range $[0, 1]$ per pixel, the lack of a robust support for single-channel image formats in video encoders (e.g. the gray-scale format *12greyle*) makes it difficult to pass the depth values directly. Furthermore, a lack of format with high-bit depth – at most 12 bits – would induce inacceptable quantization to the depth images and make them essentially unusable for compositing.

Hence, we encode the scalar field (depth image) into an image format with 3 channels and 8 bits per channel using Morton coding [22], mapping a 24-bit depth value into 3x8 bits. Here, Morton coding is much preferable to simple mapping of the high, middle, and low bytes

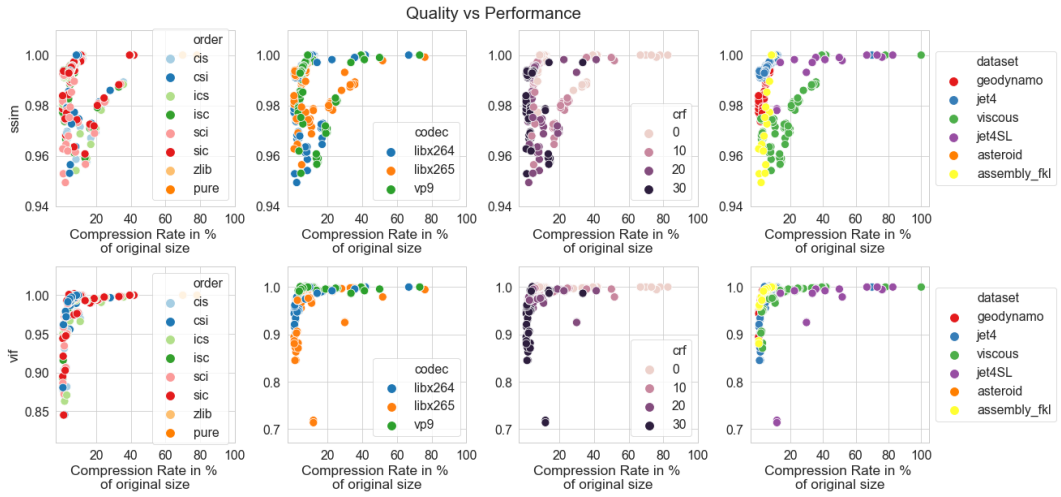


Figure 5 Comparing the results for the compression performance with the image quality. Top row: SSIM. Bottom row:VIF[24]. F.l.t.r. clustered by image ordering (s: camera **s**phere, c:**c**ycle time step, i:**i**so-value), image ordering streamlines, used video codec, constant rate factor (crf) and data set. The sweet spot is located in the top left corner, where image quality and compression rate delivers the best performance.

to three channels, since the Z-order curve underlying this coding guarantees that close-by depth values will be mapped to close-by tuples. The effect of this mapping, interpreting the three channels as red, green, and blue colors, is illustrated in Figure 6. We pass the result of this mapping to the video codec in the *yuv444p* format, as YUV is the natural color space in which all considered codecs operate and errors induced through in-codec color space conversion can be avoided. Note that video codecs typically accept various formats that represent color information at reduced resolution when compared to luminance information, such as e.g. the often used *yuv420p* format. However, we do not see an indication at this time how such sub-sampling would benefit the compression of depth images and thus only consider *yuv444p*. In general this process is applicable to any scalar field (not only depth) input on the images.

3.4 Video Compression

Following linearization and encoding, we pass the *yuv444p* image sequence directly to the video codec. As a general interface to different codecs, we employ the `ffmpeg` [14] tool, and encode using three major and broadly available video codecs:

- H.264 [26] The H.264 is using motion estimation to minimize temporal and spatial redundancies. It is classified as a block-oriented motion compensating compression technique.
- H.265 [11] The H.265 is based on the same principle than the H.264. Its main difference is the increased coding tree unit from 16×16 to 64×64, though leading in general to higher compression rates.
- VP9 [15] The VP9 codec is also a block-based format. Its main application area is for web streaming and therefor it is designed to ensure a certain bit-rate rather than a constant quality like H.264 and H.265.

Codec Parameters

In general there is a very large number of parameters that affect each encoders and allow tweaking it to different types of input. To keep our study reasonable and achieve comparable results, we opt to focus on the *constant rate factor* (**CRF**) as the central parameter that effects the amount compression for each codec, and we consider the values 0 (lossless), 10, 20, and 30 to represent different levels of compression. Typical CRF choices for natural images are in the range from 18 to 24.

All codecs furthermore support variable-rate encoding which adapts bandwidth used based on heuristics, in the interest of overall better image quality. However, as the assumptions underlying these heuristics are geared towards natural images and are not well documented, we choose not to examine this mode due to the large amount of unpredictability it induces on results. Note that the VP9 codec is primarily intended for streaming applications, and thus constant rate encoding is not its optimal mode of operation; however, we still consider it in this study to to its ubiquitous use an generally good compression / quality performance.

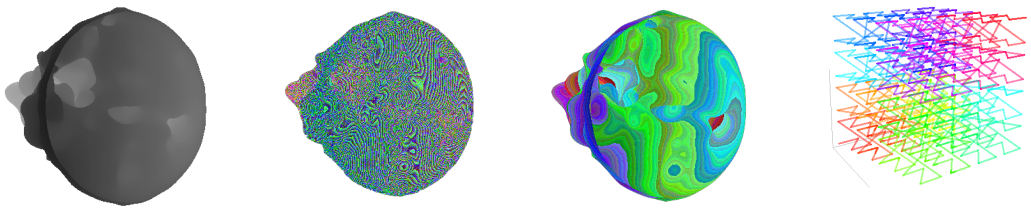
The H.264 and H.265 codecs are further able to trade off compression speed against image quality through a preset choice. We here use the *veryfast* preset to achieve the best speed. While image quality would improve with slower presets, at the expense of much longer compression times, we obtain a lower bound on image quality, which we consider to reflect real-world considerations best. A summary of the parameters steered by these presets can be found in [10]

As the output of the compression process, we obtain a single video file that represents the entire visualization depth image database.

3.5 Image Retrieval

To retrieve images, we again employ the `ffmpeg` tool to retrieve a single image from the video file in *yuv444p* format, and decode the Morton ordering.

Retrieved images are compared against the (uncompressed) ground truth images, using the metrics described in the following section.



(a) Ground Truth. **(b)** Modulo. **(c)** Morton. **(d)** Z-order curve.

Figure 6 Encoding of a 24-bit depth image (a) to 3x8 bit channels using reinterpretation as three bytes (b) or Morton coding (c). The three resulting channels are interpreted as RGB for illustration purposes. (d) shows the Z-order curve underlying the Morton coding.

4 Evaluation Metrics

For our evaluation we distinguish three major factors: Quality of the reconstructed visualizations images, compression efficiency in the sense of file size and time to retrieve an image in the original domain, and, especially for surfaces, how much the errors introduced by compression affect the compositing of two or more visualizations post hoc.

4.1 Image Quality Metrics

For our quality benchmark we choose a set of numerical and image quality measures. Typically the raw numerical metrics such as e.g. mean squared error are hard to interpret. We therefore consider quality measures for images as well. Table 1 provides an overview of all employed quality metrics.

Table 1 Used Image metrics for the quality benchmarks.

Metric	Description	Value Range (identical value)
MSE	mean squared error	$\geq 0(0)$
PSNR	peak signal to noise ratio	$\geq 0(\infty)$
MAE	mean absolute error	$\geq 0(0)$
SSIM	structural similarity index	$[0, 1](1)$
VIF	visual information fidelity measurement [24]	$[0, 1](1)$

4.2 Performance Metrics

For the performance measure we use the compression and retrieval rate. For the compression rate we use the highest z-lib compressed depth images as our base value. All values are then relative increases or decreases in percent. For the retrieval rate we randomly draw 300 frames from each video and apply the morton inversion to them to end up with depth images. We measure the time from the start of the retrieval (video already loaded) to the end of the conversion. For comparison we measure the time for 500 database calls using Paraview and the ttkCinema filter. The measures are then normalized to time to retrieve one image. We do not take speed ups from parallel computing into account, as such practical values can differ.

4.3 Compositing Benchmark

In the compositing benchmark we test how robust the video compression is for small changes in the depth value, that can lead to cluttering effects when compositing multiple surfaces in a post-processing step. This test is only done for a set of previous selected surface pairs from the Jet flow data set, representing a worst-case scenario from practice. Error is measured in the number of wrongly assigned pixel of the different surfaces to fore- and background. We do the compositing once for the ground truth depth images and once with the compressed images and then compare the two results.

5 Results

We have investigated a total of 264 encoded videos from 5 different data sets and three visualization types. In general, the video compression always achieves higher compression rates than *zlib*(see Fig.7), which we use as a general purpose compression technique and practically relevant comparison baseline. In comparison to `zlib` however, retrieval times were increased.

The H.264 codec turns out to be the most predictable, exhibiting the fastest encoding speed and generally low error. While VP9 resulted in a less overall error the encoding time is about 3 times higher for lower rate factors(see Fig.7 bottom). Regarding the constant rate

factor, as the dominant parameter for all codecs, we found out that a value around $crf = 20$ is the best trade off between compression rate and error (see Fig.5, column 4). The sweet spot is best defined as the upper left corner in Fig.5. For this parameter combinations we observe the best quality with the highest amount of compression.

Considering resulting images (Fig.1), we can observe different visual qualities for different application scenarios. With increasing CRF value the video compressor starts deleting pixels in areas with similar values. For the streamlines these areas are much smaller, which explains why the compressed file size for these is much higher.

For the surface compositing test, no significant errors can be reported. We counted the number of wrongly assigned pixels in the compositing steps. For crf = 0, no wrongly assignments were counted. For crf = [10,20,30] the maximum were 18 wrongly assigned pixels. In relation to the image size of 512 x 512 this results in a relative maximum error of 6.9×10^{-6}.

5.1 Image Quality

We can observe a predictable increase in the error rates with increasing constant rate factor. Over all settings the mean SSIM is above 0.98. Referring to Baker et.al. [5] this is still in the range were most people wont see a difference. It can be seen in Fig.1 that the outliers are nonetheless visible, but a simple smoothing may overcome this issues, as the outliers are mostly single isolated pixels. There is no significant influence of the image ordering on the resulting error rates for both iso-surface and streamline visualizations(Fig.10, row 2,3). But for the assembly use-case we can observe an effect of the ordering in the PSNR and SSIM measures. Here the quality for ordering first along the simulation index and than the camera view port (CS) achieves better results. In the assembly use-case we observe a stronger decline in image quality with increasing constant rate factor than in the other examples (see 6a). The H.265 codec is the only one which mean error is not zero for the lossless setup, resulting in the fact that it is not able to achieve a complete lossless compression (Fig.10, row 1). For higher crf values the difference between the codecs gets smaller. Regarding the image metrics the H.264 codec is overall the best performing followed by the H.265 for crf 10 and 20. For crf 30 all codecs are nearly equal. But overall the VP9 codec results in less noisy images, as it always outperforms the other codecs in the peak signal to noise ratio ($psnr$). In Fig.10 row 4, we can see that the original data set has a strong influence on the achievable image quality. Additionally the image quality for streamline visualization is more sensible to an increased rate factor than the (iso-)surface visualization (see Fig.1, row 4, blue and purple). Applying the visual information fidelity measure (vif) results in different ratings especially for the codecs and data sets (Fig.10,column 4, row 2 and 4). The mean absolute error is nearly equal among all settings and therefor no suitable measure.

5.2 Compression Rate

The compression rate of the lossless video encoding is in the order of 5 times smaller than the original uncompressed data and about 3 times smaller than zlib compression with the highest settings (Fig.7, row 1 and 2, columns 1). For lossy compression we achieve up to 40 times smaller sizes than the original data (Fig.7, row 1 and 2, column 4). The H.264 and H.265 codec achieved the highest compression results with increasing difference to VP9 with increasing rate factor (up to 3 times for $crf = 30$,see Fig.7, row 1 and 2). For the lossless compression the H.265 codec is the best performing. But we have seen in the quality benchmarks, that this codec is not able to achieve real lossless compression which leads to

an unfair advantage in this comparison. For crf 10 the H.264 codec is about 30% better than H.265 and VP which are fairly on the same level. For crf 20 the difference between H.264 and H.265 is getting smaller while VP9 does only small progress. Overall the H.264 and H.265 have a quadratic negative slope for the compression rate based on the crf value, while VP9 has a near linear slope. The first axis of image ordering has a small impact on the compression rates in the order of $2-4\%$ in absolute scale(rel. to each other $18-24\%$). The effect of the second axis is negligible. A big difference in compression rates between isosurface and streamline visualization can be observed.For the lossless setting the compression rates are near to the zlib compression. Compared to the iso-surface visualizations, the compression rate have a less steeper slope. For $crf = 30$ the compression rates are still up to 5 times higher than for iso-surfaces. The difference in the codecs also persists in this setup.

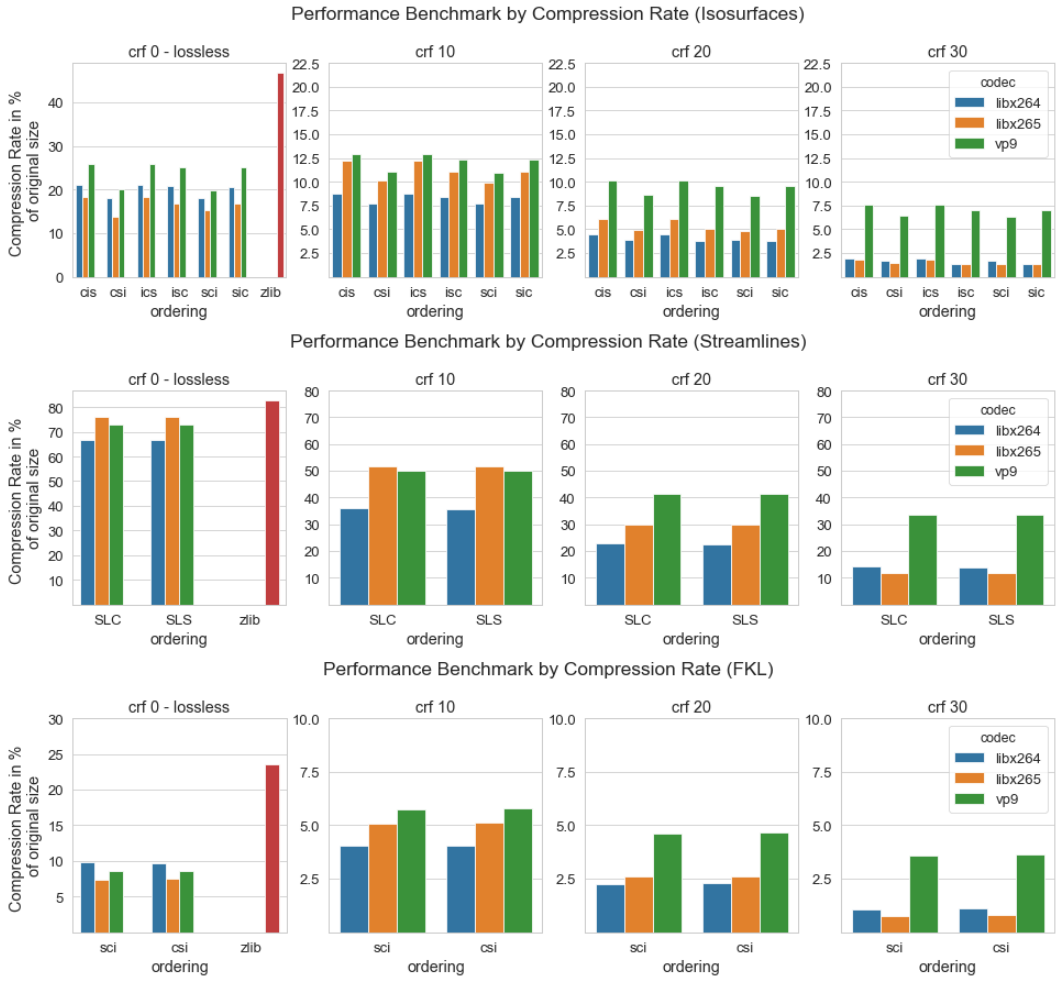


Figure 7 Results for the compression performance. Top rows: Compression rate in % of the original uncompressed image data base. grouped by image ordering for iso-surfaces(s: camera **s**phere, c:**c**ycle time step, i:**i**so-value), image ordering for streamlines (S: camera **s**phere, C:**c**ycle time step) and constant rate factors, colored by the used codec. Bottom row: Encoding time per image.

5.3 Encoding and Decoding

The encoding was performed on one node of a cluster with a Intel® Xeon® Processor E5-2640 v3. The encoding time per image was in the range of 20 ms for loss-less compression to

nearly 400 ms for the highest chosen compression rate. The rendering of the images for the database was performed on multiple machines to achieve a suitable large amount of images for the presented study. For in-situ we assume, that the data is already loaded in memory. As such only the time to extract the iso-surface and render the depth image is of relevance. In a small benchmark for images with a size of 512×512 we end up with a mean rendering time per image in the order of 72 to 230ms and a pipeline execution time for the iso-surface extraction in the order of 310 to 620ms, depending on the visualized structures complexity. The remaining driver thus is the I/O. Todays storage bandwidth is in the order of 6 -12 GB/s. The image sizes in our resulting data base are in the range of 260 KB to 295 KB. This results in 0.022 to 0.049ms per image and thus is neglectable. In a rough comparison the encoding time for the video compression adds a factor of 0.05 to 0.47 to the computation time in relation to the already performed rendering pipeline. The encoding time per image

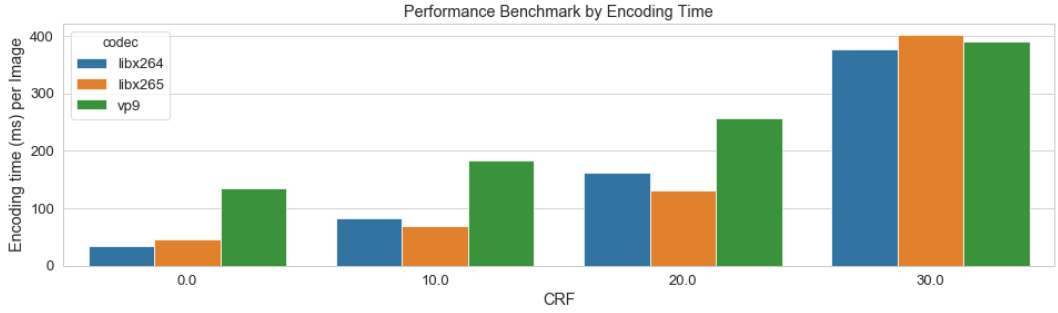


Figure 8 Results for the Encoding time per image, grouped by codec. With increasing constant rate factor (CRF) the differences between encoding times is equalizing.

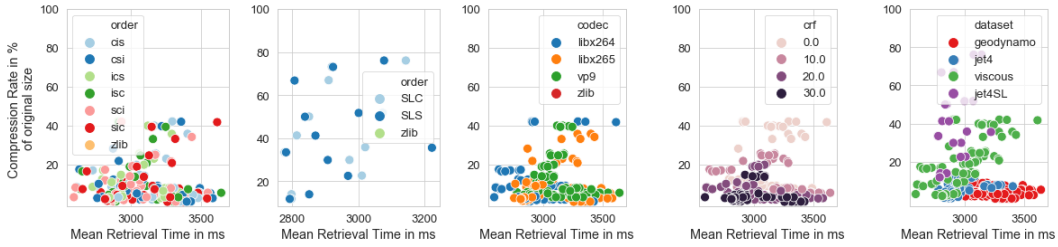


Figure 9 Comparison of compression rate and retrieval time clustered by codec, image ordering for iso-surfaces(s: camera **s**phere, c:**c**ycle time step, i:**i**so-value), image ordering for streamlines (S: camera **s**phere, C:**c**ycle time step), constant rate factor and data set.

has a nearly quadratic slope based on the crf value (Fig.7, row 3). For the codecs we can see that VP9 takes about 3 to 4 times longer, for low crf values, than H.264 and H.265. With increasing crf values the differences in encoding time between the codecs is decreasing. For lower crf values, the H.264 and H.265 codec outperform the VP9 codec. As there exist a huge numbers of hardly controllable side effects when measuring the execution time, we can make only general observations. Based on the streamline image orderings *SLC* and *SLS* compared to the orderings for the iso-surface images (*cis,csi,ics,isc,sci,sic*), we can see that the resulting compressed file size and the time to encode are negatively correlating (cor= -0.342, Pearson).

The decoding and retrieval was performed on a local desktop machine, to avoid bandwidth and network effects in the measurement. For the mean decoding or retrieval times based on random access we end up in the interval of 2.6 to 3.8 seconds. No effects for the ordering and crf value on the retrieval time can be observed (see Fig.9). There exist a slight negative correlation (cor= -0.39789, Pearson) between the size of the compressed video and the retrieval time, which means that higher compressed videos have higher retrieval times. Note that this correlation does not directly cope with the chosen crf value. For the codecs we can find some clusters which are leading to a weak order based on the retrieval times (H.264<H.265<VP9). Further Fig.9 row 5 suggests that the underlying data sets form distinguishable clusters. For the quality benchmarks, we have encountered the same effects. We found out that our measured retrieval times are about 4 to 5 times higher than using the implementations from OpenCV for frame retrieval. This implies that a faster implementation for image retrieval is possible in principle. Nonetheless the implementation from OpenCV is targeted towards the retrieval based on time stamps and not exact frame IDs. A such the retrieval is not exact, but rather chooses the next frame in a predefined range near the target.

5.4 Overall Observations

Based on five data sets with very different properties we identified that the H.264 codec delivers the overall best performance regarding image quality, compression rate and retrieval time. For encoding time it is on the same level as the slightly faster H.265 codec. Another advantage is that the H.264 codec is able to achieve true lossless compression. Regarding the investigated parameters we found that only the fastest axis chosen for linearization is significant, and the ordering of the remaining axes does not impact the results. The recent study from Baker et.al. [5] provides a guideline for choosing a good threshold on typical image metrics. To this point the most predictable one is the structural similarity metric (SSIM). For the climate model they investigated a value of 0.99995 is needed for identical visual quality, which can be achieved by choosing the constant rate factor to be 0. We determined that a value around CRF=20 delivers the best trade off between compression rate and visual quality, regarding a threshold on SSIM of 0.98 determined on medical literature by [3]. In other words, for a successful application of video compression for visualization image data bases, the only parameters a user has to determine is the fastest linearization axis along which images share the most similarity, and which quality the resulting images should satisfy. Applying video compression is not restricted to only surface-based visualization but also works for line-based visualizations, which is shown with the jet flow streamline example, but comes with decreased compression performance. In general, unsurprisingly, we observe that the compression works best for less cluttered visualizations with smooth areas and few edges.

6 Conclusions

We have provided a study concerning the applicability of lossy video compression to visualization image databases. Our findings confirm observations made by Berres et al. [7], thereby strengthening further the argument that video compression is a viable and beneficial approach leading to very good compression rates when compared to a general-purpose lossless compressor. Further, the lossy compression approach introduces only a minor image quality deterioration, and, in some case, does not lead to loss at all. We have shown that the complexity of an implementation is manageable. Furthermore, the reduced file sizes generated via video compression make generally possible the exploration of larger parameter

spaces, greatly benefiting a computational scientist when analyzing a visualization image database. Equipped with suitable meta-data, we believe that video file formats could serve as effective containers for visualization image databases generally, thus simplifying database management tasks.

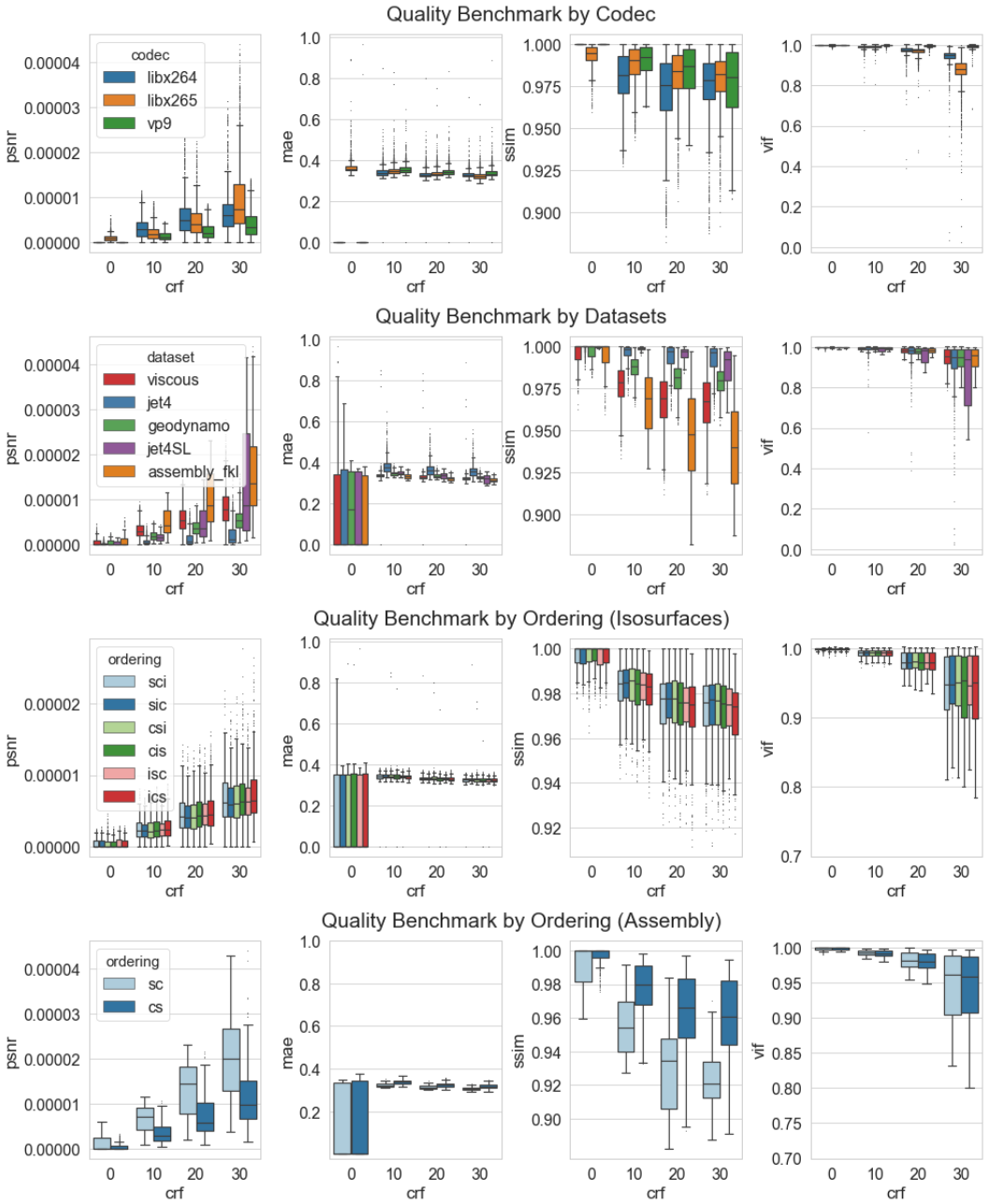


Figure 10 Results for the quality benchmark. Box-plots from top to bottom grouped by codec, image ordering for iso-surfaces(s: camera **s**phere, c:**c**ycle time step, i:**i**so-value), image ordering for streamlines (S: camera **s**phere, C:**c**ycle time step) and data set.

In general, the ordering only effects the compression with primary order parameter. Thus it is sufficient in most cases to determine the ordering axis with the highest expected similarity between subsequent images. For the image quality, we only observed an effect for the assembly

use case and there only for the SSIM and PSNR. In this setting we achieved in general higher compression rates due to the very high similarity between the images. Thus we assume that the effect of ordering on the image quality increases with higher compression rates. The use of lossy general-purpose compression techniques – i.e., techniques not primarily aimed at video compresssion – such as the ZFP [19] compressor, should be investigated. Additionally, an important aspect for many real-world applications is the ability to perform video compression and encoding in parallel for in situ visualization purposes. We believe that independent encoding of subsequences is feasible. Moreover, utilizing hardware-enabled acceleration made possible, for example, by encoding via GPUs could further improve compression performance substantially. It would also be of interest to determine a near-perfect order used to generate all in situ visualization mages of a scientific data set. If one were able to optimally define the trajectory of the "virtual camera" used to produce in situ visualization images, then standard image compression codecs would be highly effective, due to the high degree of frame-to-frame coherence for such a camera trajectory. It is planned to consider these aspects in future research.

References

1 James Ahrens, Sébastien Jourdain, Patrick O'Leary, John Patchett, David H. Rogers, and Mark Petersen. An image-based approach to extreme scale in situ visualization and analysis. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '14, pages 424–434, Piscataway, NJ, USA, 2014. IEEE Press. doi:10.1109/SC.2014.40.

2 Utkarsh Ayachit, Andrew Bauer, Berk Geveci, Patrick O'Leary, Kenneth Moreland, Nathan Fabian, and Jeffrey Mauldin. Paraview catalyst: Enabling in situ data analysis and visualization. In *Proceedings of the First Workshop on In Situ Infrastructures for Enabling Extreme-Scale Analysis and Visualization*, pages 25–29. ACM, 2015.

3 A. H. Baker, D. M. Hammerling, S. A. Mickelson, H. Xu, M. B. Stolpe, P. Naveau, B. Sanderson, I. Ebert-Uphoff, S. Samarasinghe, F. De Simone, F. Carbone, C. N. Gencarelli, J. M. Dennis, J. E. Kay, and P. Lindstrom. Evaluating lossy data compression on climate simulation data within a large ensemble. *Geoscientific Model Development*, 9(12):4381–4403, 2016. doi:10.5194/gmd-9-4381-2016.

4 Allison H Baker, Dorit M. Hammerling, and Terece L. Turton. Evaluating Image Quality Measures to Assess the Impact of Lossy Data Compression Applied to Climate Simulation Data. *Computer Graphics Forum*, 2019.

5 Allison H. Baker, Dorit M. Hammerling, and Terece L. Turton. Evaluating Image Quality Measures to Assess the Impact of Lossy Data Compression Applied to Climate Simulation Data. *Computer Graphics Forum*, 2019.

6 Janine C. Bennett, Hank Childs, Christoph Garth, and Bernd Hentschel. In Situ Visualization for Computational Science (Dagstuhl Seminar 18271). *Dagstuhl Reports*, 8(7):1–43, 2019. doi:10.4230/DagRep.8.7.1.

7 Anne S. Berres, Terece L. Turton, Mark Petersen, David H. Rogers, and James P. Ahrens. Video Compression for Ocean Simulation Image Databases, 2017. doi:10.2312/envirvis.20171104.

8 Tim Biedert and Christoph Garth. Contour tree depth images for large data visualization. In *Eurographics Symposium on Parallel Graphics and Visualization, Cagliari, Sardinia, Italy, May 25 - 26, 2015.*, pages 77–86, 2015. doi:10.2312/pgv.20151158.

9 Tim Biedert, Peter Messmer, Thomas Fogal, and Christoph Garth. Hardware-accelerated multi-tile streaming for realtime remote visualization. In *EGPGV 2018: Eurographics Symposium on Parallel Graphics and Visualization, Brno, Czech Republic, June 4, 2018*, pages 33–43, 2018. doi:10.2312/pgv.20181093.

10 Joey Blake. FFmpeg preset files. Contribute to joeyblake/FFmpeg-Presets development by creating an account on GitHub, May 2019. original-date: 2010-09-12T13:01:27Z. URL: `https://github.com/joeyblake/FFmpeg-Presets`.

11 Benjamin Bross, Mauricio Alvarez-Mesa, Valeri George, Chi Ching Chi, Tobias Mayer, Ben H. H. Juurlink, and Thomas Schierl. Hevc real-time decoding. In *Optics & Photonics - Optical Engineering + Applications*, 2013.

12 David Ellsworth, Bryan Green, Chris Henze, Patrick Moran, and Timothy Sandstrom. Concurrent visualization in a production supercomputing environment. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):997–1004, 2006.

13 H. Hagen F. Claus, F.-A. Rupprecht. Online simulation considering production uncertainties to improve assembly quality. In *NAFEMS World Congress*. NAFEMS, 2019. URL: `https://www.nafems.org/publications/resource_center/nwc_19_356/`.

14 FFmpeg Developers. Ffmpeg. URL: `https://ffmpeg.org/`.

15 Adrian Grange, Peter de Rivaz, and Jonathan Hunt. VP9 Bitstream & Decoding Process Specification v0.6, 2016.

16 M. Hummel, R. Bujack, K. I. Joy, and C. Garth. Error estimates for lagrangian flow field representations. In *Proceedings of the Eurographics / IEEE VGTC Conference on Visualization: Short Papers*, EuroVis '16, pages 7–11, Goslar Germany, Germany, 2016. Eurographics Association. `doi:10.2312/eurovisshort.20161153`.

17 Akira Kageyama and Tomoki Yamada. An approach to exascale visualization: Interactive viewing of in-situ visualization. *Computer Physics Communications*, 185(1):79–85, 2014.

18 S. Li, N. Marsaglia, C. Garth, J. Woodring, J. Clyne, and H. Childs. Data reduction techniques for simulation, visualization and data analysis. *Computer Graphics Forum*, 37(6):422–447, 2018. `doi:10.1111/cgf.13336`.

19 Peter Lindstrom. Fixed-rate compressed floating-point arrays. *IEEE transactions on visualization and computer graphics*, 20(12):2674–2683, 2014.

20 Jonas Lukasczyk, Eric Kinner, James Ahrens, Heike Leitte, and Christoph Garth. A View-Approximation Oriented Image Database Generation Approach. *IEEE 8th Symposium on Large Data Analysis and Visualization(LDAV), 2018*, page 11, 2018.

21 K. Ma. In situ visualization at extreme scale: Challenges and opportunities. *IEEE Computer Graphics and Applications*, 29(6):14–19, November 2009. `doi:10.1109/MCG.2009.120`.

22 Guy M. Morton. *A computer oriented geodetic data base and a new technique in file sequencing*. International Business Machines Company, 1966.

23 Patrick O'Leary, James Ahrens, Sébastien Jourdain, Scott Wittenburg, David H. Rogers, and Mark Petersen. Cinema image-based in situ analysis and visualization of mpas-ocean simulations. *Parallel Computing*, 55:43–48, 2016. Visualization and Data Analytics for Scientific Discovery. `doi:10.1016/j.parco.2015.10.005`.

24 Hamid R. Sheikh and Alan C. Bovik. Image information and visual quality. In *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages iii–709. IEEE, 2004.

25 Bong-Soo Sohn, Chandrajit Bajaj, and Vinay Siddavanahalli. Volumetric video compression for interactive playback. *Computer Vision and Image Understanding*, 96(3):435–452, 2004.

26 Gary J. Sullivan, Pankaj Topiwala, and Ajay Luthra. The h.264/avc advanced video coding standard: overview and introduction to the fidelity range extensions. In *SPIE Optics + Photonics*, 2004.

27 Julien Tierny, Guillaume Favelier, Joshua A. Levine, Charles Gueunet, and Michael Michaux. The Topology ToolKit. *IEEE transactions on visualization and computer graphics*, 24(1):832–842, 2018.

28 Anna Tikhonova, Carlos D. Correa, and Kwan-Liu Ma. Explorable images for visualizing volume data. In *IEEE Pacific Visualization Symposium PacificVis 2010, Taipei, Taiwan,March 2-5, 2010*, pages 177–184, 2010. `doi:10.1109/PACIFICVIS.2010.5429595`.

29 Zhou Wang, Eero P. Simoncelli, and Alan C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee, 2003.

30 J. Woodring, S. Mniszewski, C. Brislawn, D. DeMarle, and J. Ahrens. Revisiting wavelet compression for large-scale climate data using jpeg 2000 and ensuring data precision. In *2011 IEEE Symposium on Large Data Analysis and Visualization*, pages 31–38, October 2011. doi:10.1109/LDAV.2011.6092314.

Finite Element Simulation Combination to Predict the Distortion of Thin Walled Milled Aluminum Workpieces as a Result of Machining Induced Residual Stresses

Daniel Weber[1]
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Benjamin Kirsch
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Christopher R. Chighizola
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Julianne E. Jonsson
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Christopher R. D'Elia
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Barbara S. Linke
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Michael R. Hill
Department of Mechanical and Aerospace Engineering, University of California Davis, CA, USA

Jan C. Aurich
Institute for Manufacturing Technology and Production Systems,
Technische Universität Kaiserslautern, Germany

Abstract

Machining induced residual stresses (MIRS) are a main driver for distortion of monolithic thin walled aluminum workpieces. A typical machining process for manufacturing such geometries for the aerospace industry is milling. In order to avoid high costs due to remanufacturing or part rejection, a simulation combination, consisting of two different finite element method (FEM) models, is developed to predict the part distortion due to MIRS. First, a 3D FEM cutting simulation is developed to predict the residual stresses due to machining. This simulation avoids cost intensive residual stress measurements. The milling process of the aluminum alloy AA7050-T7451 with a regular end mill is simulated. The simulation output, MIRS, forces and temperatures, is validated by face milling experiments on aluminum. The model takes mechanical dynamic effects, thermomechanical coupling, material properties and a damage law into account. Second, a subsequent finite element simulation, characterized by a static, linear elastic model, where the simulated MIRS from the cutting model are used as an input and the distortion of the workpiece is calculated, is presented. The predicted distortion is compared to an additional experiment, where a 1 mm thick wafer was removed at the milled surface of the aluminum workpiece. Furthermore, a thin walled component that represents a down scaled version of an aerospace component is manufactured and its distortion is analyzed.

The results show that MIRS could be forecasted with moderate accuracy, which leads to the conclusion that the FEM cutting model needs to be improved in order to use the MIRS for a correct prediction of the distortion with the help of the linear elastic FEM model. The linear elastic model on the other hand is able to predict the part distortion with higher accuracy when using measured data instead of MIRS from the cutting simulation.

[1] corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 11; pp. 11:1–11:21

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases Machining induced residual stresses, distortion, Finite element method simulation

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.11

Funding The authors would like to thank the German Research Foundation (DFG) and the National Science Foundation (NSF) for the financial support within the project AU 185/64-1 "NSF DFG Collaboration to Understand the Prime Factors Driving Distortion in Milled Aluminum Workpieces". UC Davis received funding from the NSF under Award No. 1663341 (CMMI). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

[1]Naming of specific manufacturers is done solely for the sake of completeness and does not necessarily imply an endorsement of the named companies nor that the products are necessarily the best for the purpose.

1 Introduction

The use of thin walled structural monolithic aluminum components like rib type shaped geometries is established in the aerospace industry due to their unique properties. These parts are characterized by advantages like low assembly times, high production efficiency, low production costs and good material properties like high overall-strength-to-weight ratio, high specific strength and corrosion resistance [22]. A typical machining process for such rib type shaped geometries is milling. Hereby up to 90 % of the material is removed [3]. As these components typically are long, up to 14 m, and thin walled, down to 2 mm, distortion is a common problem [3]. High costs due to remanufacturing or rejection of the parts are a result [30]. It is known that the main cause of those distortions are residual stresses (RS) [22]. They are defined as the internal stresses locked in a body, where force and torque equilibrium prevail and no thermal gradients appear [30]. In this context two types of RS are from greater interest. First, there are the RS which exist in the blank material because of former processes like heat treatments (e.g. quenching) prior to machining. These stresses are called initial bulk residual stresses (IBRS), because they appear throughout the entire part-thickness [22]. Typically, aircraft parts are machined from stress relieved aluminum alloys, e.g. AA7050-T7451, to reduce the IBRS and increase the mechanical strength [12]. The second type of RS which contribute to the distortion are the machining induced residual stresses (MIRS). They are introduced into the material during the machining process and their penetration depth is limited to a shallow surface layer of the part.

Finite element method (FEM) simulations are often used to predict the part distortion due to MIRS of milled thin walled monolithic aluminum workpieces. Mostly two different groups of FEM approaches are recognizable in the literature. Group I models are characterized by calculating the distortion of the workpiece due to the temperatures and forces of the milling process [1, 27, 33]. Furthermore, group I models considered the IBRS, milling path and clamping forces. The cutting process was hereby simplified in a way that the material removal was modeled by the element deletion of the elements along the milling path. No tool workpiece interaction was simulated. Forces and temperatures were applied to the mesh nodes while the part geometry was constrained. The temperature and force data used as input were obtained by different methods. Rai et al. e.g. calculated the forces analytically using Oxley's predictive machining theory [27]. Temperatures were determined by using the heat dissipation rate. Bi et al. used a FEM process model in DEFORM-3D[1] to simulate the

forces [1]. Furthermore, the heat was derived from the shear energy. It was assumed that the shear energy, which was calculated analytically with the help of the forces from the process model, was transformed completely into heat. Tang et al. used a FEM cutting simulation in DEFORM-3D[1] to obtain both, temperatures and forces [33]. All group I models contain a second simulation step, where the constraints were changed to allow the workpiece to deform, producing a prediction of distortion but not MIRS.

Group II models are characterized by static, linear elastic FEM simulations, where the RS are known and used as an input to predict the part distortion [3, 13, 15, 24, 25, 28]. This modelling approach only takes the final residual stress state of the machined part into account, able to consider both, MIRS and IBRS. Mostly, the stresses are applied to the finished part shape, whereas some models also consider the material removal by element deletion. The re-equilibrium of the stresses causes the workpiece to distort. The MIRS required for the simulation were mostly obtained by measured data [3, 13, 15, 25, 28] or analytical models [11, 17, 34]. Another method of gaining information on MIRS, and saving costly RS measurements, are 2D, respectively orthogonal cutting [15, 23, 26, 29], and 3D cutting simulations [16, 21, 24, 36]. 3D cutting simulations are more useful because they are able to predict the entire stress tensor including shear stresses, which are essential stresses when it comes to predicting distortion of thin walled parts [2, 7]. There are not many 3D cutting simulations that predict MIRS on aluminum workpieces available in the literature. Li et al. predicted the MIRS for milling aluminum alloy Al2024-T3 by using a cutting simulation in Third Wave Systems AdvantEdge[1] [21]. The material behavior was defined by a power law with strain hardening and thermal softening. Furthermore, a self-adaptive meshing technique was applied. As a simulation result, only the maximum compressive RS were presented and compared to measurements. However, in order to predict the part distortion accurately the full depth profile is required. Ma et al. e.g. achieved a prediction of the MIRS depth profile from milling a steel alloy by using cutting simulations in Third Wave Systems AdvantEdge[1] [24]. The Johnson-Cook material model was used. A large simulation error was stated for the comparison of distortion measurement and simulation result. Furthermore no shear stresses were presented, although they are an important factor of distortion due to MIRS [2, 7]. No validation of the MIRS in form of residual stress measurements was done.

In the present research FEM simulations of group II were investigated to predict the distortion of milled thin walled aluminum workpieces due to MIRS. The MIRS were obtained by a FEM cutting simulation for the aluminum alloy AA7050-T7451. Experiments were carried out as validation. Forces, temperatures and RS were compared to experimental data. Furthermore, the distortion of a thin wafer and a thin walled workpiece geometry was measured and compared to the model outcome. Additionally, a simulation environment in Matlab[1] was developed, which embeds both, cutting and distortion FEM simulations.

2 Methods

2.1 Experimental set up

2.1.1 Force and machining induced residual stress measurements

Down milling was carried out on a 5-axis DMG Mori DMU 70 CNC[1] machine with cemented carbide end mills. An end mill is a typical tool for machining of aerospace aluminum alloys. The tool properties can be found in Table 1.

In order to minimize the IBRS, the AA7050-T74 samples come in a stress relieved condition (T7451). The samples were cut from a large slab of stress relieved aluminum measuring 1250x1250x102 mm^3. Six blocks of material measuring 660x206x102 mm^3 were

Table 1 Tool properties.

tool properties	Kennametal[1] F3AA1200AWL
type	regular end mill (EM)
diameter	12 mm
tool holder	HSK-A 63
material	cemented carbide 10 % Co, 0.6 % Cr, 89.4 % WC max. grain size 3 µm
number of flutes z	3
helix angle	45°
cutting edge radius	-
length	76 mm
coating	-

Table 2 Machining parameters.

machining parameter	
cutting speed v_c	200 m/min
feed per tooth f_z	0.2 mm
depth of cut a_p	3 mm
width of cut a_e	4 mm
coolant	dry
direction	down milling
simulation input	SI-Unit
spindle speed $n = \frac{v_c}{\pi d}$	88 rounds/s
feed rate $v_f = f_z\ z\ n$	0.0531 m/s

saw cut from the large slab. These blocks were cut again into 15 individual plates measuring 206x102x28.5 mm^3, where the 206x102 mm^2 face was milled. The tool movement was along the negative x-direction (206 mm dimension) with respect to the sample coordinate system (see Fig. 1a).

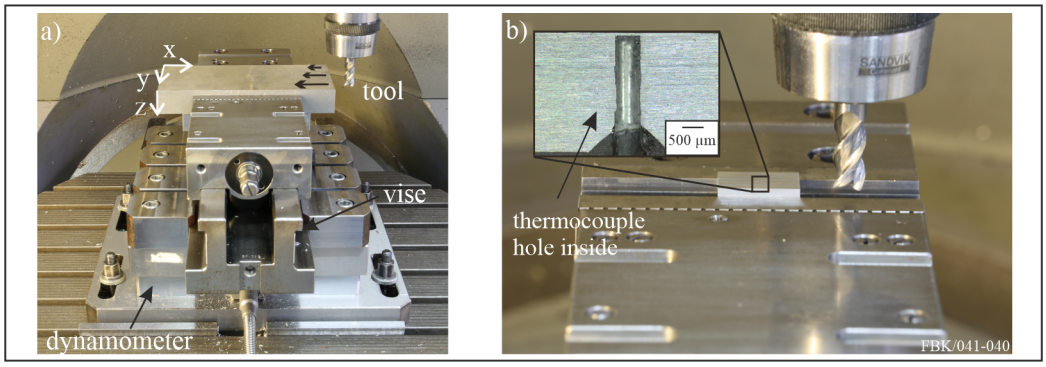


Figure 1 Experimental set up for MIRS, force (a) and temperature measurements (b).

The orthogonal feed in y-direction is along the 102 mm dimension. The machining parameters can be found in Table 2. The aluminum workpiece was clamped in a conventional vise with 125 mm long jaws and a clamping force of approximately 15 kN was applied. 5.5 mm of the workpiece height protruded prior to cutting. The process was monitored by recording forces using a piezoelectric dynamometer (Kistler Type 9255[1]) with a sampling rate of 15 kHz. One surface layer was removed, which resulted in 25 tool passes with a constant width of cut of 4 mm. The last 2 mm were removed in an additional pass, which was not included in the measurements. The three orthogonal forces (F_x, F_y, F_z), where F_x is the force in negative feed direction, F_y in orthogonal feed direction and F_z is the passive force (see Figure 1a), were analyzed. Three samples (ID 1, ID 2, ID 3) were machined to investigate the reproducibility of the MIRS.

The MIRS were measured with the hole-drilling (HD) technique following the ASTM E837-13a standard [14] on the milled 206x102x28.5 mm^3 samples. The HD technique uses a rosette strain gage to measure three components of strain while material is removed in fine increments [6]. Three components of residual stress (σ_{xx}, σ_{yy}, τ_{xy}) as a function

of cut depth are resolved from strain as a function of cut depth [14]. HD was chosen, because previous research by the authors showed that data from HD are most consistent with machining induced distortion for AA7050-T7451 parts [2]. Furthermore, multiple measurements are necessary to investigate the reproducibility of the MIRS resulting from one machining condition. Therefore, three HD measurements were conducted on the milled surface (206x102 mm^2) of each aluminum workpiece. This resulted in nine MIRS depth profiles in total. The position and depth schedule can be found in Fig. 2. The final depth of the hole was measured afterwards to compute the offset (= measured final depth – intended final depth), which was then used for a depth correction of the data (offset is then added to all depths) [2]. The analysis of the stresses is done by interpolating the nine independent measurements to the respective depth schedule in Fig. 2 and calculating the average stresses at each depth and their standard deviation at each depth.

a)

increment in mm	depth in mm
0	-
0.0127	0.0127
0.0127	0.0254
0.0127	0.0381
0.0127	0.0508
0.0127	0.0635
0.0127	0.0762
0.0127	0.0889
0.0127	0.1016
0.0127	0.1143
0.0127	0.1270
0.0254	0.1524

increment in mm	depth in mm
0.0254	0.1778
0.0254	0.2032
0.0254	0.2286
0.0254	0.2540
0.0254	0.2794
0.0508	0.3302
0.0508	0.3810
0.0508	0.4318
0.0508	0.4826
0.0508	0.5334
0.0508	0.5842

b)

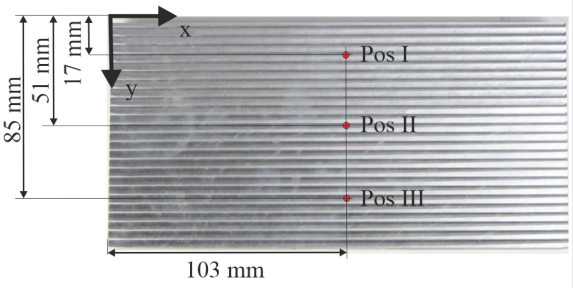


Figure 2 Depth schedule (a) and position (b) of hole-drilling measurements.

2.1.2 Temperature experiment

The temperature measurements were carried out in separate experiments on smaller AA7050-T7451 samples (30x20x9 mm^3, see Figure 1b). The sample dimensions had to be decreased in comparison to the previous experiments in order to manufacture the thermocouple hole with an accurate depth. The orientation of the tool movement was the same compared to prior experiments. The feed (negative x-) direction was along the 30 mm dimension. The sample was clamped in the vice with 4 mm of the sample height protruded prior to cutting. Thermocouples type K (1KI10TDT-40-4000MS)[1] with a diameter of 1 mm were inserted from the bottom side in the middle of the sample. The nominal distance from their end face to the cutting face was 100 µm. The temperatures at a depth of 100 µm, resulting from one cut, are not influenced by the sample size. To investigate the reproducibility, three samples (ID T1, ID T2, ID T3) were machined. The sampling rate of the temperature measurements was set to 2 kHz. The value of the maximum temperature was calculated by the arithmetic mean of 200 temperature values around the total temperature maximum of each sample (range ± 0.05 s).

2.1.3 Wafer distortion experiment

The wafer distortion experiment was designed to show the distortion potential due to the MIRS and validates the simulation model. A cube of the size 25x25x25.5 mm^3 was removed out of the larger machined sample ID 2 (206x102x25.5 mm^3) by wire electric discharge

machining (EDM). A second EDM cut was necessary to cut off a 1 mm thick wafer, including the machined surface. The distortion of the wafer was measured on the wire EDM surface (bottom) with the help of a laser profilometer. Points with a 0.2 mm spacing across the 25x25 mm^2 face of the wafer were scanned with a distance of 1 mm towards the edges.

2.1.4 Feature distortion experiment

The feature distortion experiment also served as a validation for the simulation model. Here a more complex geometry was machined. A component that resembles a down scaled version of a thin walled rib type component, which is typically used in the aerospace industry, was chosen and hereafter called feature sample. It is a small rib type component with one rib in the middle, surrounded by two pockets with a wall thickness of 3 mm (see Fig. 3).

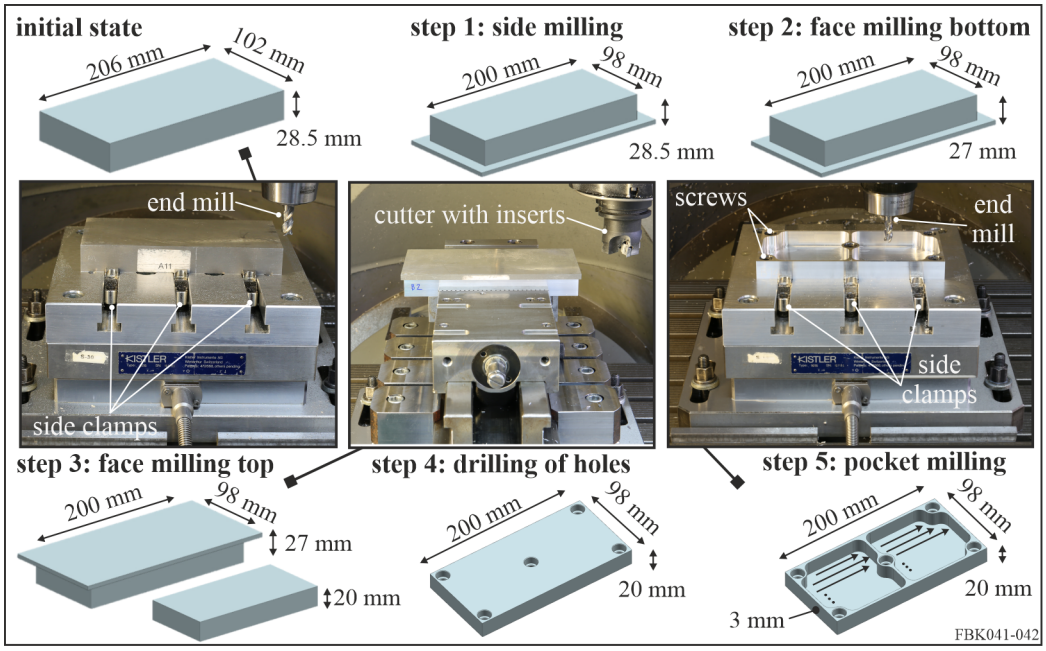


Figure 3 Machining steps of feature sample manufacturing.

The feature sample was also machined out of one of the above mentioned stress relieved AA7050-T7451 blocks with the dimensions 206x102x28.5 mm^3. Different milling operations were conducted to achieve the final geometry of the sample (200x98x20 mm^3), where about 84 % of the initial material was removed. The pockets were machined with the same machining parameters, milling strategy (zig strategy: paths from left to right, see step 5 in Fig. 3) and tool type (EM), investigated previously, where the MIRS were known. Face milling was carried out with a second tool, a cutter with indexable inserts (IDX). The tool properties can be found in Table 3. Here, the machining parameters were chosen to hardly induce any residual stresses. So, the assumption was made that only the MIRS resulting from machining the pockets contributes to the distortion of the workpiece due to MIRS.

The machining parameters can be found in Table 4. Two different clamping devices, a conventional vise and side clamps, were used (see Fig. 3). The following machining steps were executed (see Fig. 3):

1. Side milling (roughing and finishing) with EM in side clamps: 200x98x28.5 mm^3
2. Face milling bottom surface with IDX in side clamps: 200x98x27 mm^3

Table 3 Tool properties (cutter with indexable inserts).

tool properties	Sandvik[1] R590-050HA6-11M
type	cutter with indexable inserts (R590-110504H-NL H10)[1]
diameter	50 mm
tool holder	HSK-A 63
material	cemented carbide
number of inserts z	2
major cutting edge angle	90°
cutting edge radius	0.4 mm
functional length	71 mm
coating	-

3. Reclamping sample upside down in vise, face milling (IDX) top surface: 200x98x20 mm^3
4. Drilling of holes with 9 mm drill
5. Reclamping in side clamps and screws and milling of pockets (zig strategy with alternating milling of pocket layers) with EM

Table 4 Machining parameters for manufacturing the feature sample.

machining parameter	side milling (EM) step 1	face milling (IDX) step 2 and 3	pocket milling (EM) step 5
cutting speed v_c	450 m/min	730 m/min	200 m/min
feed per tooth f_z	0.055 mm	0.2 mm	0.2 mm
depth of cut a_p	5x 4.4 mm (roughing) 22 mm (finishing)	1.5 mm (bottom); 5x 1.4 mm (top)	3 mm
width of cut a_e	2.5 mm (roughing) 0.5 mm (finishing)	40 mm	4 mm

Side clamps and screws were chosen for clamping while milling the pockets to prevent the bottom surface from distortion before the workpiece was completely finished. Finally, the distortion of the bottom surface at the backside was measured with the coordinate measuring machine Tesa micro hite 3D DCC[1] with a repeatability limit (ISO MPE-p) of 3.5 µm. A spacing of the measured points of 2 mm was chosen with a distance of 1 mm to the edges.

2.2 Simulation model

The simulation combination to predict distortion of milled thin walled aluminum workpieces due to MIRS consists mainly of two different FEM models, embedded in an Matlab[1] environment (see Fig. 4). The two FEM models are a 3D, dynamic, elastic-plastic cutting simulation and a 3D, static, linear elastic FEM model [37]. The cutting model simulates the tool workpiece interaction and predicts the MIRS due to the machining parameters (input). With the help of the linear elastic model the MIRS from the previous model are applied at the workpiece to forecast the part distortion. Both FEM models are modelled and calculated in ABAQUS[1]. The link between the two models, extracting the MIRS depth profile and applying it automatically to the boundary layer of the workpiece of the second model, is handled by a developed Matlab[1] script (see Fig. 4).

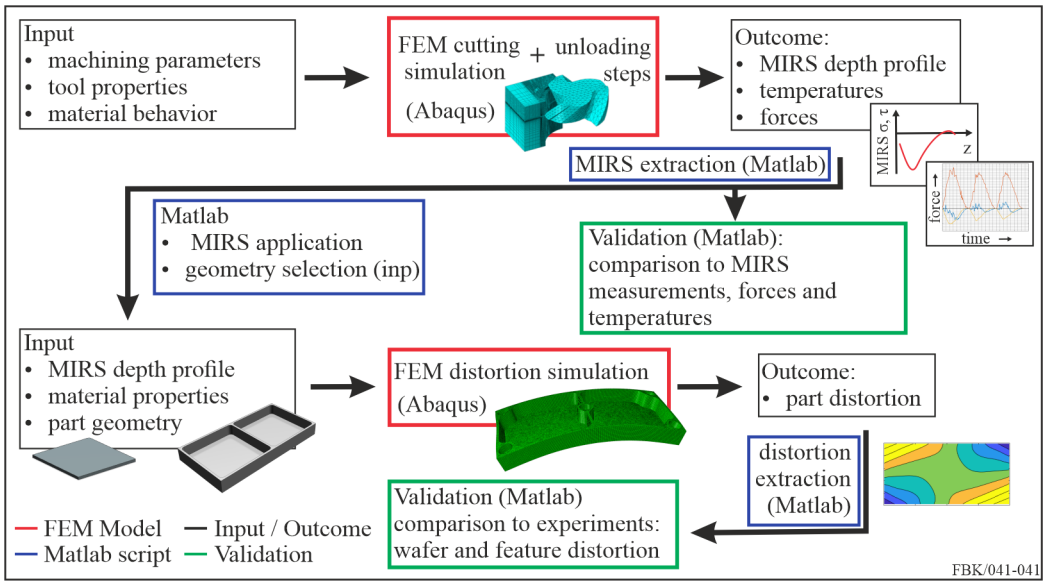


Figure 4 Simulation combination: Two FEM models embedded in Matlab[1] environment.

2.2.1 3D dynamic FEM cutting model

The required information on the MIRS resulting from milling are obtained by a 3D FEM cutting simulation, which is set up in ABAQUS[1]. An explicit, dynamic, elastic-plastic approach is chosen. An explicit solver is used, because the modelling of the complex contact problem requires the application of an explicit solver. This is very efficient for simulating highly non-linear problems involving large localized deformations, high strain rates and changing contact conditions as those experienced in the cutting process [38]. As input data the machining parameters feed rate v_f and rotation speed n are given in SI-Units (see Table 2). Only one rotation of the tool, which lasts 0.0113 s, was computed due to large computational times. The tool is assumed as a rigid body, neglecting wear. Rigid body is an acceptable assumption, because of the significantly high elastic modulus of the cemented carbide tool and the resulting low elastic deflection of it compared to the large plastic deformation of the aluminum workpiece. Neglecting wear is an acceptable assumption, because only one rotation of the tool is simulated. A thermomechanical analysis is chosen to consider both, mechanical and thermal effects. Furthermore, the tool is shortened to a length of 5 mm. The workpiece is a 10x8x15 mm^3 block with a circular cutout of the size of the tool radius (6 mm). The workpiece used in the cutting simulation is way smaller than a typical rib type component or the workpiece used for the milling experiments (see Fig. 5) to reduce the number of elements and save computational costs, because cutting simulations need a high computational time.

The mesh consists of 760,919 eight-node thermally coupled brick, trilinear displacement and temperature elements with reduced integration (C3D8RT). A fine mesh in the cutting zone, especially in the near surface region, is necessary to resolve the MIRS depth profile accurately. Here the dimensions of the elements are 50x20x20 µm^3 (see Fig. 6). The workpiece is hold in place with the help of fixture boundary conditions (BC), which set all degrees of freedom to 0 at the faces marked in Fig.6 [32]. Furthermore, the movement in x- and y- direction of the nodes on the arc surface are constrained, because in reality the circular cutout would be filled with material. The tool consists of 29,217 four-node thermally coupled tetrahedron, linear displacement and temperature elements (C3D4T) with a global element size of 0.5 mm and a finer mesh near the cutting edge (see Fig. 6).

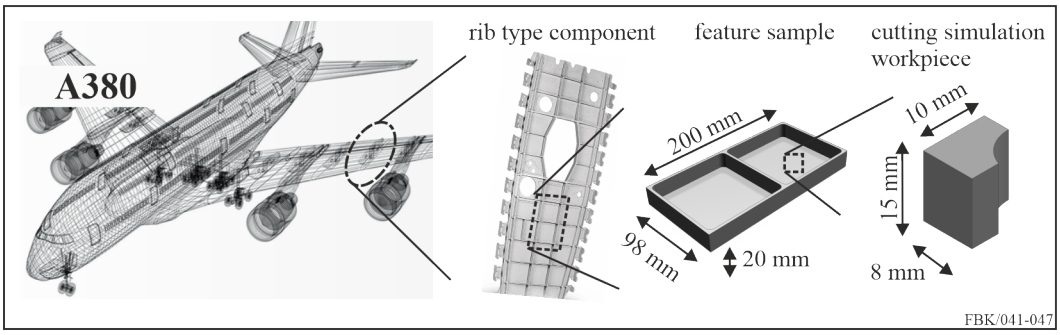


Figure 5 Different dimensions of real assembly, part and workpieces used for simulations [9, 10, 37].

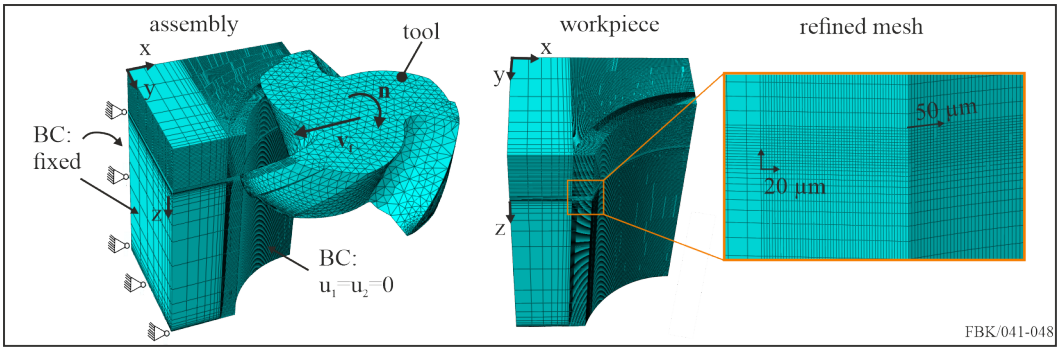


Figure 6 Mesh of tool and workpiece.

The material behavior of the workpiece is described via the Johnson-Cook strain rate dependence material model, where the yield stress at nonzero strain rate $\bar{\sigma}$ of the material is defined according to Eq.(1) [19].

$$\bar{\sigma} = \left[A + B\left(\bar{\varepsilon}^{pl}\right)^{n}\right]\left[1 + C\,\ln\left(\frac{\dot{\bar{\varepsilon}}^{pl}}{\dot{\epsilon}_0}\right)\right]\left(1 - \frac{T - T_{trans}}{T_{melt} - T_{trans}}\right) \tag{1}$$

In Eq.(1) $\bar{\varepsilon}^{pl}$ is the equivalent plastic strain, $\dot{\bar{\varepsilon}}^{pl}$ is the equivalent plastic strain rate and $\dot{\varepsilon}_0$ is the reference plastic strain rate, which is commonly assumed to be 1 [32]. The temperatures T, T_{melt} and T_{trans} are the prevailing temperature, the melting temperature and the transition temperature. A, B, C, m and n are material parameters. n and m are the strain hardening coefficient, respectively the thermal softening coefficient. A, B and C describe the yield stress behavior. The Johnson-Cook material parameters for AA7050-T7451 used in the present research are listed in Table 5 [35]. The start of material damage is modelled via the Johnson-Cook damage initiation criterion, which is a special case of the ductile criterion [32]. Damage happens when the criterion for damage initiation is met (Eq.2).

$$\omega_D = \int \frac{d\bar{\varepsilon}^{pl}}{\bar{\varepsilon}^{pl}_D(\eta, \dot{\varepsilon}^{pl})} = 1, \tag{2}$$

where ω_D is a state variable, η is the stress triaxiality $\eta = -p/q$ with pressure stress $p = -tr\,\boldsymbol{\sigma}$ (Cauchy stress tensor $\boldsymbol{\sigma}$) and Mises equivalent stress $q = \sqrt{\frac{3}{2}\boldsymbol{\sigma}^D \cdot \boldsymbol{\sigma}^D}$ (stress deviator $\boldsymbol{\sigma}^D$) and the equivalent plastic strain at the onset of damage $\bar{\varepsilon}^{pl}_D$, which is defined according to

Eq.(3) [19].

$$\bar{\epsilon}_D^{pl} = \left[d_1 + d_2 \exp\left(d_3 \frac{p}{q}\right)\right]\left[1 + d_4 \ln\left(\frac{\dot{\bar{\varepsilon}}^{pl}}{\dot{\varepsilon}_0}\right)\right]\left(1 + d_5 \frac{T - T_{trans}}{T_{melt} - T_{trans}}\right) \tag{3}$$

In Eq.3 d_1 to d_5 are the Johnson-Cook damage parameters (see Table 6) [20].

Table 5 Johnson-Cook material parameter for aluminum alloy AA7050-T7451 [35].

A	B	C	m	n	T_{melt}	T_{trans}	$\dot{\varepsilon}_0$
490 MPa	207 MPa	0.005	1.8	0.344	600 °C	20 °C	1

Table 6 Johnson-Cook damage parameter [20].

d_1	d_2	d_3	d_4	d_5	T_{melt}	T_{trans}	$\dot{\varepsilon}_0$
-0.77	1.45	-0.47	0	1.6	600 °C	20 °C	1

The damage evolution is characterized by a damage evolution law based on the linear displacement of the elements. The effective plastic displacement $\bar{u}_f^{pl}$ is described via the evolution Eq.(4).

$$\dot{\bar{u}}^{pl} = L\dot{\bar{\epsilon}}^{pl}, \tag{4}$$

where L is defined as the characteristic length of the element [32]. The damage variable is a function of the equivalent plastic displacement $d = d(\bar{u}^{pl})$. A linear evolution of the damage variable is chosen, Eq.(5).

$$\dot{d} = \frac{L\dot{\bar{\epsilon}}^{pl}}{\bar{u}_f^{pl}} = \frac{\dot{\bar{u}}^{pl}}{\bar{u}_f^{pl}} \tag{5}$$

Final failure is defined when the effective plastic displacement $\bar{u}^{pl}$ reaches the specified effective plastic displacement at point of failure ($\bar{u}^{pl} = \bar{u}_f^{pl}$), which is here defined by the smallest element length of 20 µm. At this point element deletion takes place and allows for the separation of the material, respectively the chip formation. Element deletion is realized by setting the stiffness matrix at the affected element to 0 ($d = 1$ fully degraded). The entire material behavior is summarized by Figure 7a. Besides the Johnson-Cook material parameters, more mechanical and thermal material parameters, e.g. density and Young's modulus, for workpiece and tool need to be defined (see Table 7). Heat is caused by the deformation of the material involving high inelastic strains [31]. This is modeled by a volumetric heat flux defined via the inelastic heat fracture β option in ABAQUS[1]. It is assumed that 90 % of the energy caused by deformation is converted into thermal energy [23]. Furthermore, it is defined, that all of the dissipated energy due to friction is released as heat and a surface film condition enables the heat convection from the model surfaces to the surrounding air [31].

The contact between workpiece and tool was modelled via general contact interaction. Therefore, a surface with all workpiece elements (exterior and interior entities) was defined. Coulomb friction was applied. In the literature there is no agreement on the friction coefficient between a cemented carbide tool and an aluminum alloy workpiece found [5]. Here a friction coefficient µ of 0.3 was chosen [18]. The simulation was run on the high-performance computer

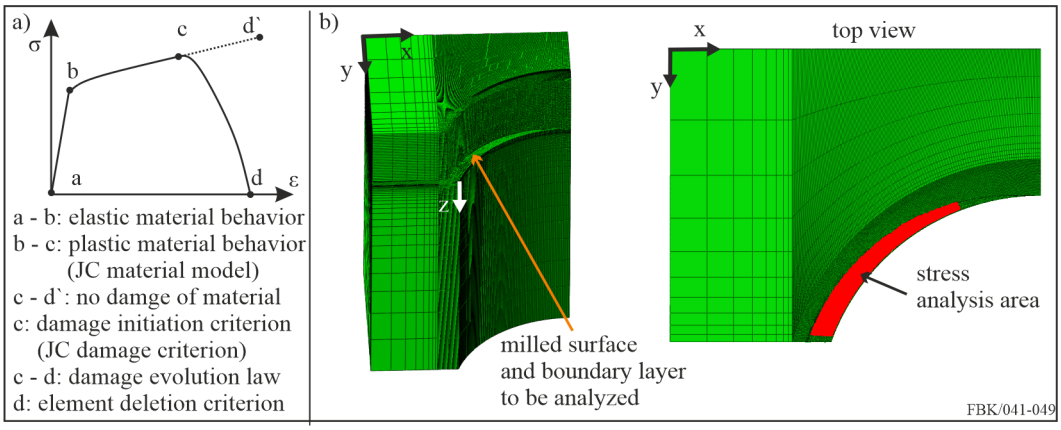


Figure 7 Typical uniaxial stress-strain response of a metal specimen (according to [32]) (a) and deformed workpiece with highlighted residual stress analysis area (b).

Table 7 Material parameters for workpiece and tool.

material properties	workpiece	tool
density ρ in kg/m^3	2,830	15,000
Young's modulus E in MPa	71,700	640,000
Poisson ratio ν	0.33	0.22
thermal expansion coefficient α in 1/K	25.5E-6	6E-6
thermal conductivity coefficient k in W/(m K)	157	55
specific heat capacity c_p in J/kg°C	860	240
inelastic heat fracture β	0.9	-
film coefficient h in W/(m^{2}K)	80	-

"Elwetritsch" at the TU Kaiserslautern with a computational time of 7 days. Besides the cutting simulation itself, unloading steps after the cutting process are necessary to guarantee that the residual stresses are in equilibrium. Two unloading steps were executed in a second simulation: The deformed mesh of the workpiece along with the information on stresses and temperatures were imported (via Matlab[1]) in a new ABAQUS[1] FEM model, which simulated the process of cooling down the workpiece to room temperature and releasing the fixture BC. At the end the MIRS were analyzed with a developed Matlab[1] script. Therefore, the stresses at all nodes of the area marked in Fig. 7b, at the milled surface and in the boundary layer of it, were extracted and averaged for each depth. Furthermore, the standard deviation was computed for each depth and discussed together with the residual stress results in section 3.3.

2.2.2 3D Static, linear elastic FEM model

A static, linear elastic finite element model was set up in ABAQUS[1] to simulate the distortion due to the MIRS. The MIRS (σ_{xx}, σ_{yy}, τ_{xy}) extracted from the cutting model and the measured MIRS (as validation) respectively, were implemented as an input and the distortion was calculated, after equilibrium had been set. Plane stress with $\sigma_{zz} = 0$, $\sigma_{xz} = 0$, and $\sigma_{yz} = 0$ was assumed. The geometries were, according to the wafer, a 25x25x1 mm^3 thin plate (see Fig. 8a), and the thin walled feature sample (see Fig. 8b).

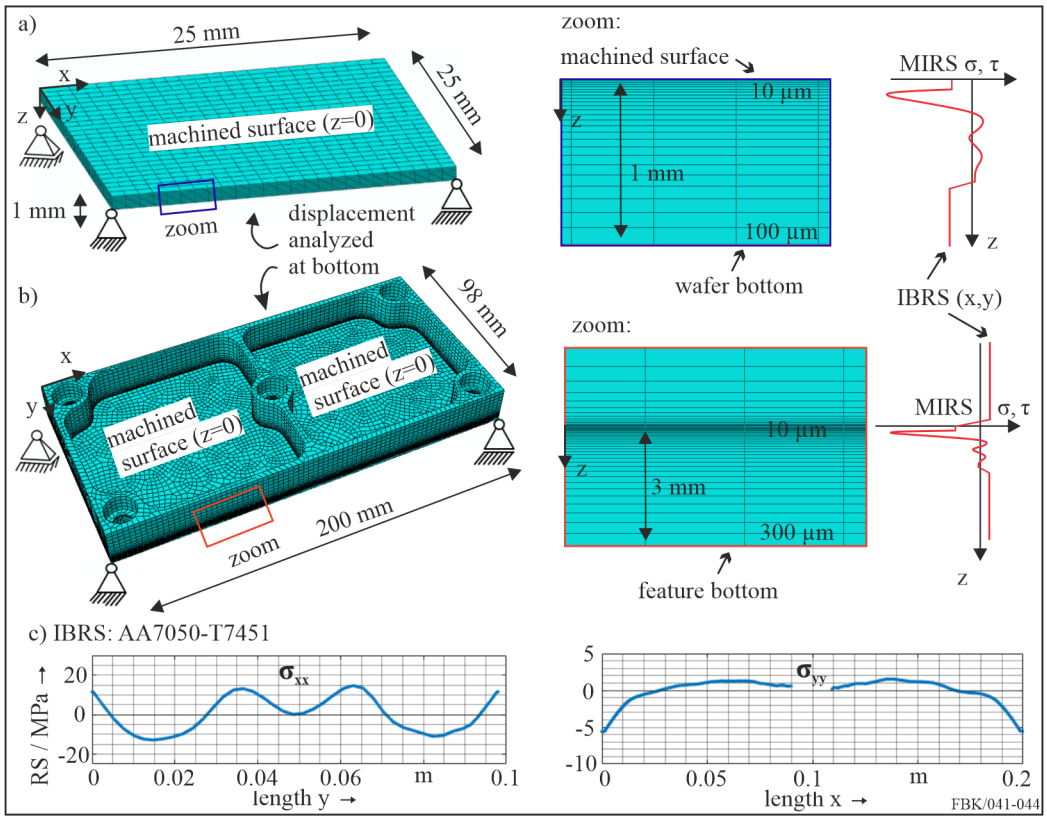


Figure 8 Wafer (a) and feature linear elastic FEM model (b), and measured IBRS (c).

The z-direction corresponds to the depth of the MIRS. The MIRS were linearly interpolated over depth z at the element centroids and applied as an initial condition on the entire milled surface of the wafer and on the bottom surface of the pockets of the feature sample. For depths smaller than the first interpolated point the first interpolated MIRS was used (see Fig. 8). For depths greater than the last measured depth the IBRS, which were left over in the blank material after the stress relief process, are used. They were measured using the slitting method prior to milling (see Fig. 8c) [8]. The IBRS depend on the x- and y- position. The stresses applied at the centroids were linearly interpolated accordingly to their coordinates. The mesh of both geometries consisted of eight-node brick elements (C3D8) with 62,500 elements in total for the wafer and 152,936 elements for the feature sample. The global size of the elements for the wafer geometry was set to 500 µm. Due to the larger geometry of the feature sample the global size of the elements was set to 2.5 mm. There were 25 elements in z-direction of the wafer with the smallest size of 10 µm at the machined surface and bigger elements at the bottom face (100 µm) (see Fig. 8), in order to precisely resolve the residual stresses near the surface and still reduce the total number of elements for calculation time reasons. Due to the larger thickness of the bottom of the feature sample (3 mm) compared to the wafer the smallest size of 10 µm increased to the bottom to 300 µm. Both parts, wafer and feature, were constraint by the 3-2-1 constrain principle, which avoided rigid body motion, but enabled a free distortion of the bodies [1]. Linear elastic material behavior with a Young's modulus of 71,700 MPa and a Poisson ratio ν of 0.33 was given. After equilibrium was calculated, the displacement at the bottom was analyzed and compared to the wafer and feature experimental distortion.

3 Results and discussion

First, the results of the FEM cutting simulation are discussed in terms of force, temperature, MIRS and compared to experimental data (see section 3.1-3.3). Second, the outcome of the subsequent FEM model, which predicts the distortion, is discussed and compared to the wafer and feature distortion experiments (see section 3.4.1, 3.4.2).

3.1 Force analysis

The forces F_x, F_y and F_z of the simulation and the experiments can be found in Fig. 9a. One rotation of the tool (t=0.013 s), which means three cutting edge engagements, is illustrated. The forces of the experiment were extracted from the force signal of pass 14 of sample ID 1.

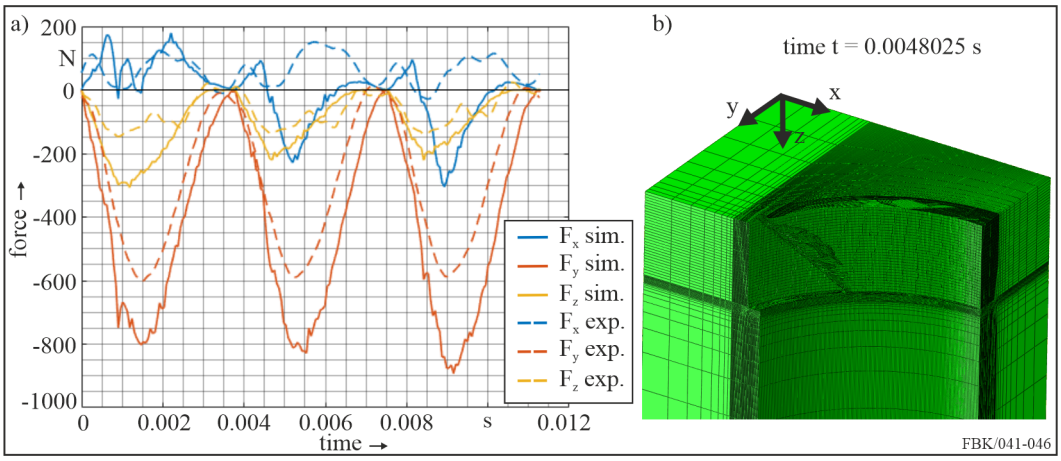


Figure 9 Force comparison of simulation and experiment (a) and simulated chip formation (b).

The three peaks for each force component correspond to the engagement of each of the three flutes. Forces F_y are the highest in magnitude, because this is the direction where the primary cutting happens (see Fig. 9b). The experimental force $F_{y_{exp}}$ shows a maximum magnitude of 600 N. The simulation force $F_{y_{sim}}$ overcomes this maximum. Its highest magnitude is about 800 N for the fist two engagements and 865 N for the last engagement. A similar trend occurs for the forces in z-direction. The maximum magnitude of $F_{z_{sim}}$ is higher than measured in experiments (200 N compared to 140 N). F_x shows positive and negative values because of the superposition of the cutting of the cutting edge and the pushing of the major flank face in the other direction. The comparison of forces F_x shows that the trend of the experimental force signal is only matched for the first engagement, where a similar magnitude is reached. The forces of the second and third cut are mostly negative, which indicates that the effect of pushing of the major flank face is dominating in the simulation. The magnitude of the different simulated forces for the different engagements differs from each other, because of multiple reasons. First there is more material removed for the first cut compared to the others due to the geometry of the chip. The first chip does not represent the actual chip form, since the initial geometry has a cutout of a perfect circle with a radius of 6 mm. Second, the simulation does not account for a continuous chip. Often elements of the chip were deleted due to the element deletion criterion, which tears the chip into several smaller parts. However, it can be stated that forces of simulation and experiment agree to a certain level. Differences in experimental and simulated forces might be because of the absence of elastic effects from the tool being in contact with the workpiece

and machining vibrations, which occur during real machining. Furthermore, the parameters used for describing the plastic material behavior and damage of the material are literature values, which may not suit best for the used aluminum alloy.

3.2 Temperature analysis

The temperature of the simulation is analyzed at the node N_T, which is located on the third cross-section of the undeformed chip, 110 µm underneath the milled surface (see Fig. 10a). After 0.008249 s the third flute's cutting edge just passed the node N_T and the maximum temperature $T_{spot_{max}}$ of 40.05 °C is reached (see Fig. 10b). The other two local maxima in the temperature profile resulted from the first two engagements of the tool.

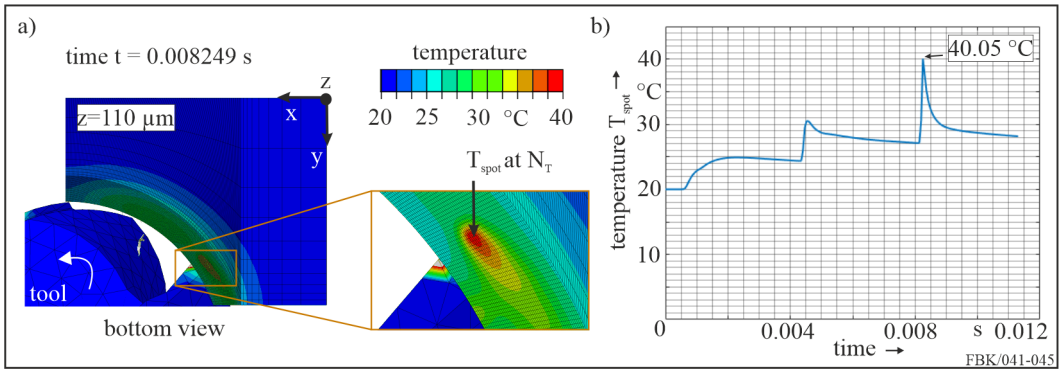


Figure 10 Position of temperature analysis (a) and the temp. profile over time at node N_T (b).

The maximum temperature of the milling experiments can be found in Table 8. The max. temperature is reached shortly after the tool moved over the thermocouple. The average value of three maximum temperatures (ID T1, ID T2, ID T3) is 32.2 °C. The temperatures of both simulation and experiment are on a similar level and agree well. One reason why the experimental temperature is slightly lower could be that the sampling rate of 2 kHz of the thermocouple is not enough to resolve the temperature profile in the same way the simulation does (15 kHz). Furthermore, the heat conduction in reality between the thermocouple face and the workpiece has to be considered, although a heat-conducting paste was used.

Table 8 Experimental measured maximum temperatures.

maximum temperature sample ID T1	maximum temperature sample ID T2	maximum temperature sample ID T3	average max. temperature
33.6 °C	31.7 °C	31.4 °C	32.2 °C

3.3 Machining induced residual stress analysis

The simulated and measured MIRS are compared in Fig. 11. All MIRS follow a square root-shaped stress profile with a pronounced maximum of residual stress (MaxRS), which is defined here as the highest absolute value of RS including compressive (negative) RS as well. This depth profile is typical for MIRS from milling [22].

The simulated normal MIRS at shallow depths (first extracted measured depth) are tensile, followed by a layer of compressive RS, whereas the shear RS are consistently negative. After the MaxRS is reached, the stress profiles of all three components increase and get to

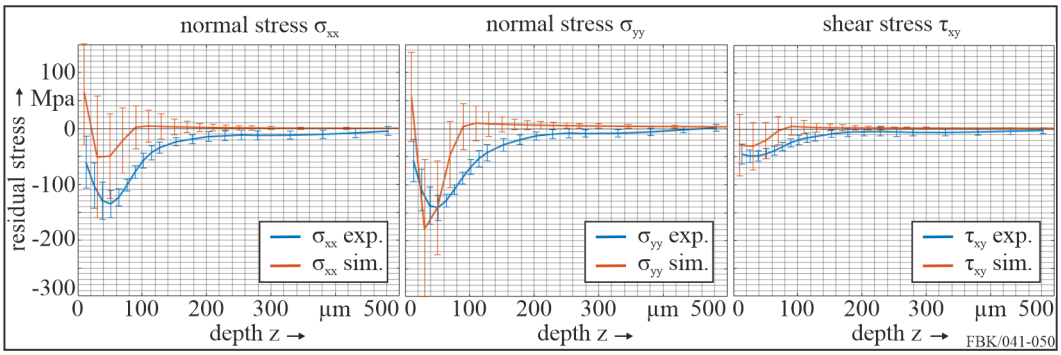


Figure 11 Comparison of MIRS from simulation and experiment.

0 MPa at about 90±20 µm (normal stresses) and 70±20 µm (shear stress). The measured MIRS show a similar behavior, except that the first measured normal RS is compressive and the MIRS are driven deeper into the material. The experimental MIRS reach 0 MPa at about 230±20 µm. The simulated MaxRS in y-direction and its depth are on a similar level compared to measurements. MaxRS_{yy} of the simulation is -180±124 MPa at 30 µm compard to -140±23 MPa at 50 µm in experiments. The simulated RS in x-direction are smaller in magnitude. The simulated MaxRS_{xx} is -50±76 MPa at a depth of 50 µm compared to -136±24 MPa at a depth of 50 µm in the experiments. The behavior of different MaxRS for y- and x-direction could be explained by the forces, which also show differences in their maximum magnitude (see section 3.4.2). In addition the simulated forces in y-direction match measured forces better than in x- direction. The simulated shear RS are lower in magnitude with MaxRS_{xy} -31±42 MPa at 30 µm compared to measurements -49±12 MPa at 25 µm. It is noticeable that the standard deviations (STD) of the simulated MIRS are high, especially for shallow depth smaller than 50 µm. The STD of the first extracted depth (10 µm) is the biggest, because the majority of the elements of the first layer were deleted. The few remaining ones were not deleted properly during the first cut, leaving distorted elements behind. This resulted not only in a high STD, but also in an area of high tensile RS near the surface (see Fig. 12). That means that the RS of the first extracted point are not representative and will be excluded for further use in section 3.4.1. The reason for the high STD of the following depth layers is the inhomogeneous distribution of RS (see Fig. 12) in the cutting area due to the influences of boundary conditions and chip formation process [36] (see Fig. 12). The differences between simulation and experiment might be explained by the absence of the run out of the tool (movement of the tool over the milled surface) in the simulation, which does not happen because only one rotation of the tool is simulated. This also leads to the undesirable effect that the stress analysis area is close to the workpiece edges. Also, the effect of multiple machining passes in orthogonal feed direction on the MIRS is not covered by the simulation. Furthermore, the material behavior and friction modelling is based on literature values, which might not be accurate for the used material. Besides the mesh, the stable time increment and boundary conditions could influence the results.

In general, it can be stated that the typical depth profile with compressive RS is represented by the simulation on average. Especially the profile of shear RS and the depth of MaxRS agree well with measured data. Differences are found for the MaxRS and for the thickness of the RS layer with measured data.

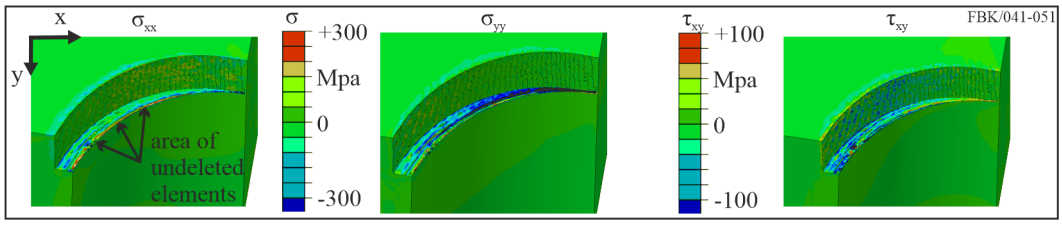


Figure 12 Deformed workpiece with residual stress distribution σ_{yy}.

3.4 Distortion analysis

The distortion analysis is divided into two sections. First the distortion of the simple wafer geometry is analyzed and discussed. After that, the distortion of the more complex feature geometry is investigated. For both geometries, the 3D linear elastic FEM model is first validated by using the measured residual stress data as an input and comparing the results to measured distortion. Next the MIRS from the cutting simulation are used as an input to analyze how the differences of MIRS from the cutting simulation compared to the measured MIRS effect the distortion prediction.

3.4.1 Wafer distortion analysis

Figure 13 shows the color maps of the measured wafer distortion at the bottom (extracted from sample ID 2) and the wafer distortion predicted by the linear elastic FEM model using the averaged MIRS from hole drilling measurements and the left over low IBRS as an input. The color maps mimic looking down at the top milled surface with positive distortion in the z-direction (into the surface). The machined surface becomes convex ($\cap$-shaped) due to the compressive MIRS at the top milled surface. Its maximum distortion is found at the top left (0 mm, 0 mm) and bottom right corner (25 mm, 25 mm). This kind of twisted distortion is due to the shear RS, which cause a torsional moment in addition to the bending moment caused by the normal RS [2, 7]. Therefore, mainly the shear RS are responsible for the maximum distortion on this diagonal. The contour plots of the measured and simulated distortion match. Both show the previously described convex-shaped distortion on a similar level. The line plots in Fig. 13 represent the distortion on the two diagonals and highlight, that the maximum distortion of the experiment is not fully achieved by the simulation. 82 % of the maximum distortion found on the diagonal \ is reached by the simulation. The differences in distortion might be explained by the measurement inaccuracies of the HD technique and the RS induced by the EDM cutting, although the cutting parameters were chosen to minimize the stress induced during cutting.

In general, it can be concluded that this simulation type is able to predict the distortion of simple flat geometries due to MIRS from milling and left over IBRS with a good accuracy.

Figure 14 shows the measured distortion of the wafer compared to the predicted distortion when using the MIRS from the cutting simulation instead of the hole drilling data and the measured left over low IBRS. It can be seen that the simulated distortion shape is convex as well, but with lower magnitude and a rotated shape of the low distortion region at the center compared the to the measured distortion. The maximum distortion still is found on top left (0 mm, 0 mm) and bottom right corner (25 mm, 25 mm). But now only 33 % of the maximum measured distortion is reached by the simulation. The lower magnitude of distortion could be explained by the lower magnitude and penetration depth of simulated MIRS from the cutting simulation compared to experimental data (see Fig. 11). The reason for the rotated distortion behavior is the difference of the magnitude of simulated normal MIRS (MaxRS_{yy} = -180 MPa and MaxRS_{xx} = -50 MPa).

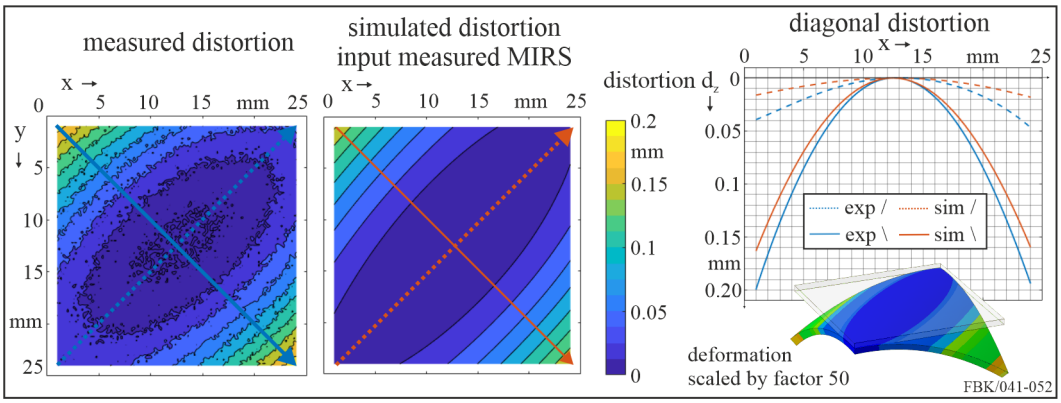


Figure 13 Measured and simulated wafer distortion with measured MIRS as input for simulation.

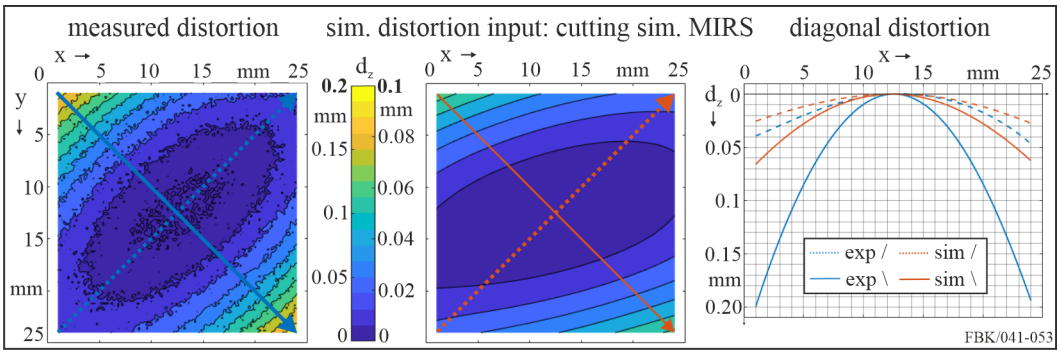


Figure 14 Measured and simulated wafer distortion with MIRS from cutting simulation as input.

3.4.2 Feature distortion analysis

Figure 15 shows the comparison of the measured distortion of the feature sample (at the backside) and the predicted distortion of the linear elastic FEM model due to the measured MIRS and low left over IBRS. The color maps mimic looking down at the milled surfaces of the pockets with positive distortion in the z-direction. Similar to the wafer distortion the maximum distortion can be found near the top left (0 mm, 0 mm) and bottom right corner (200 mm, 98 mm). In contrast to the wafer distortion the lowest level is not in the middle of the sample, which results in a X-shaped distortion. This is because of the more complex geometry of the feature sample with one rib and walls, which improve the stiffness of the geometry. Overall, the simulation is able to predict the shape of distortion, although it overestimates the maximum distortion in the corners by 132 % of the measured value (see Fig. 15). The differences in the measured and predicted distortion might be because only the MIRS at the bottom surface of the pockets and the left over IBRS are considered. Whereas in reality the machining of other surfaces such as the backside (step 2) or side walls (step 5) of the feature sample also induce low RS, which contribute to part distortion.

Figure 16 shows the measured distortion of the feature sample (at the backside) compared to the predicted distortion when using simulated MIRS and the measured left over low IBRS. Similar to the wafer distortion it can be stated that the MIRS from the cutting simulation as input lead to lower distortion than in reality (mind different scales in Fig. 16). Only 41 % of the maximum distortion found on the diagonal \ is reached by the simulation. But in general the simulated shape of distortion matches the experiments.

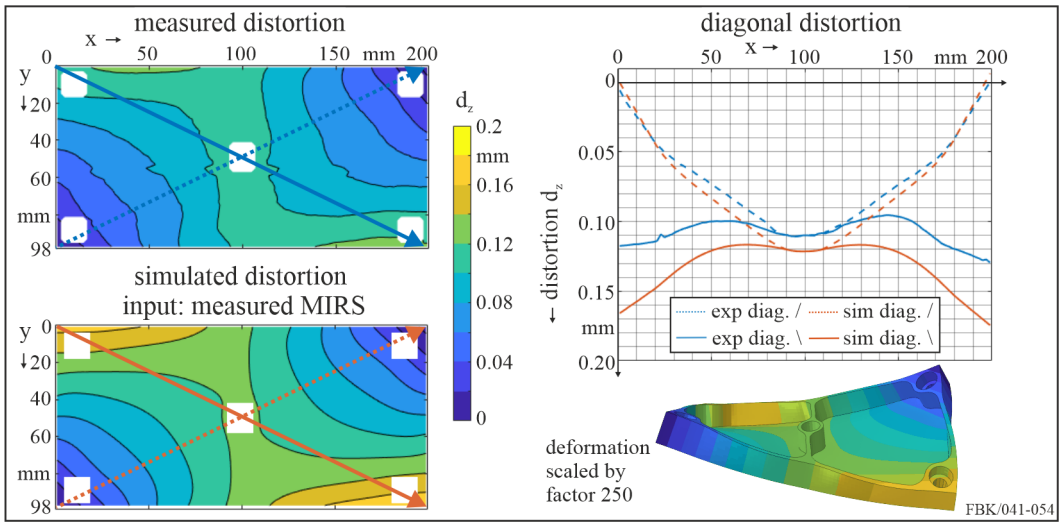


Figure 15 Measured and simulated feature distortion with measured MIRS as input.

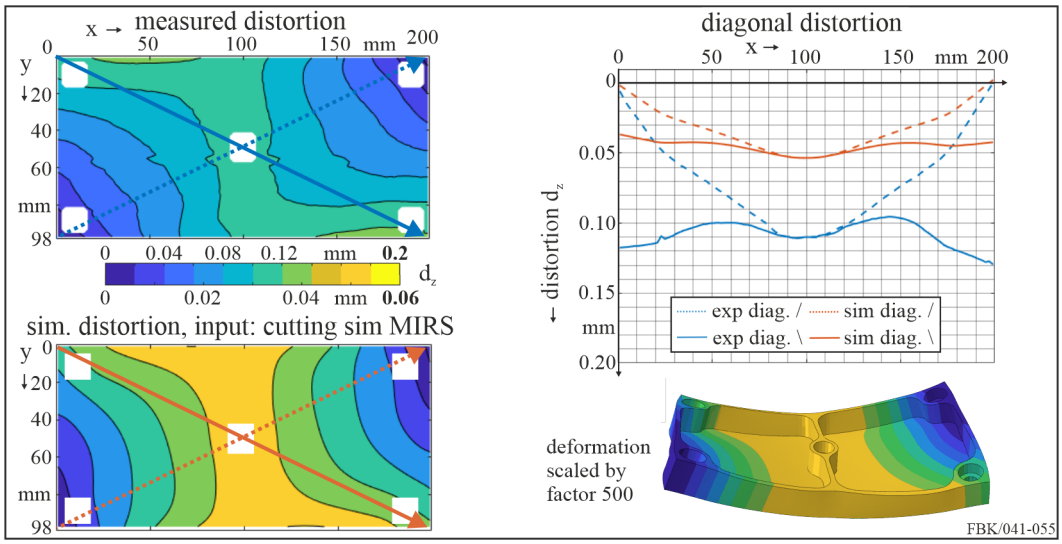


Figure 16 Measured and simulated feature distortion with MIRS from cutting sim. as input.

4 Conclusion and Outlook

The 3D, dynamic, elastic-plastic cutting FEM model was able to predict forces, temperatures and the MIRS profile with moderate accuracy. The depth profile of the MIRS with its typical square root-shape and the level of RS in general could be predicted qualitatively. But it also showed big deviations from reality in the form of lower penetration depth and lower values of the MaxRS (except in y-direction).

The linear elastic FEM model was able to predict the part distortion for simple (wafer) and more complex (feature) geometries qualitatively when using the measured MIRS and the left over low IBRS as input. In order to improve the simulation, MIRS at different surfaces need to be considered.

When combining both simulation models, the part distortion due to the simulated MIRS could not be predicted correctly, yet. Although the shape of distortion matches qualitatively, large deviations on the level of distortion are found, because of the deviations from the simulated MIRS from reality. This shows, that accurate MIRS are critical when used to predict distortion due to milling. Therefore, the cutting simulation needs some improvements. For example a finer resolution of the mesh, not only in the cutting zone, and the simulation of multiple tool rotations, to create a bigger MIRS analysis area and consider the run out of the tool, are necessary and will be investigated in future works. Furthermore, a deeper investigation on the used material models, e.g. a material flow stress curve optimization analysis similar to the approach presented by Eisseler et al. [4] could help to improve the results. Besides, the influence of the stable time increment, boundary conditions and friction model need to be investigated in detail. Moreover, the effect of different machining parameters used as an input for the cutting simulation on the MIRS and their distortion will be investigated.

References

1 Y. Bi, Q. Cheng, H. Dong, and Y. Ke. Machining distortion prediction of aerospace monolithic components. *Journal of Zhejiang University-SCIENCE A*, 10(5):667–668, 2009. doi:10.1631/jzus.A0820392.

2 C.R. Chighizola, C.R. D'Elia, D. Weber, B. Kirsch, J.C. Aurich, B.S. Linke, and M.R. Hill. Intermethod Comparison and Evaluation of Measured Near Surface Residual Stress in Milled Aluminum. *Currently under review in Expermintal Mechanics*, Submitted April 2020.

3 B. Denkena and S. Dreier. Simulation of residual stress related part distortion. In *New Production Technologies in Aerospace Industry - Proceedings of the 4th Machining Innovations Conference, Hannover, Germany*, pages 105–113, 2014. doi:10.1007/978-3-319-01964-2_15.

4 R. Eisseler, K. Drewle, K.C. Grötzinger, and H.C. Möhring. Using an inverse cutting simulation-based method to determine the johnson-cook material constants of heat-treated steel. In *Procedia CIRP 77 - Proceedings of the 8th CIRP Conference on High Performance Cutting*, pages 26–29, 2018. doi:10.1016/j.procir.2018.08.198.

5 K. Giasin, A. Hodzic, V. Phadnis, and S. Ayvar-Soberanis. Assessment of cutting forces and hole quality in drilling al2024 aluminium alloy: experimental and finite element study. *International Journal of Advanced Manufacturing Technolgy*, 87:2041–2061, 2016. doi:10.1007/s00170-016-8563-y.

6 P. Grant, J. Lord, P. Whitehead, and A.T. Fry. The Application of Fine Increment Hole Drilling for Measuring Machining-Induced Residual Stresses. *Applied Mechanics and Materials*, 3-4:105–110, 2005. doi:10.4028/www.scientific.net/AMM.3-4.105.

7 M. Gulpak, J. Sölter, and E. Brinksmeier. Prediction of shape deviations in face milling of steel. In *Procedia CIRP 8 - Proceedings of the 14th CIRP Conference on Modelling of Machining Operations*, pages 15–20, 2013. doi:10.1016/j.procir.2013.06.058.

8 M.R. Hill. The slitting method. In *Schajer GS (ed). Practical residual stress measurement methods. Wiley, West Sussex, UK*, 2013.

9 DMG Mori Homepage. https://de.dmgmori.com, last accessed 2020/08.

10 Mitsubishi Carbide Homepage. http://www.mitsubishicarbide.com, last accessed 2020/08.

11 X. Huang, X. Zhang, and H. Ding. An enhanced analytical modelof residual stress for peripheral milling. In *Procedia CIRP 58 - Proceedings of the 16th CIRP Conference on Modelling of Machining Operations*, pages 387–392, 2017. doi:10.1016/j.procir.2017.03.245.

12 X.M. Huang, J. Sun, and J.F. Li. Effect of initial residual stress and machining-induced residual stress on the deformation of aluminium alloy plate. *Journal of Mechanical Engineering*, 612:131–137, 2015. doi:10.5545/sv-jme.2014.1897.

13 X.M. Huang, J. Sun, and J.F. Li. Finite element simulation and experimental investigationon the residual stress-related monolithic component deformation. *International Journal of Advanced Manufacturing Technolgy*, 77:1035–1041, 2015. doi:10.1007/s00170-014-6533-9.

14 ASTM International. E837-13a Standard Test Method for Determining Residual Stresses by the Hole-Drilling Strain-Gage Method. West Conshohocken, PA:. *ASTM International*, 2013. doi:10.1520/E0837-13A.

15 S. Jayanti, D. Ren, E. Erickson, S. Usui, T. Marusich, K. Marusich, and H. Elanvogan. Predictive modeling for tool deflection andpart distortion of large machined components. In *Procedia CIRP 12 - Proceedings of the 8th CIRP Conference on Intelligent Computation in Manufacturing Engineering*, pages 37–42, 2013. doi:10.1016/j.procir.2013.09.008.

16 X. Jiang, Y. Zhu, Z. Zhang, M. Guo, and Z. Ding. Investigation of residual impact stress and its effects on the precision during milling of the thin-walled part. *International Journal of Advanced Manufacturing Technolgy*, 97:9–12, 2018. doi:10.1007/s00170-018-1941-x.

17 Z. Jiang, Y. Liu, L. Li, and W. Shao. A novel prediction model for thin plate deflections consideringmilling residual stresses. *International Journal of Advanced Manufacturing Technolgy*, 74:37–45, 2014. doi:10.1007/s00170-014-5952-y.

18 X. Jin. Mechanics and dynamics of micro-cutting process. In *Dissertation, The University of British Columbia-Vancouver*, 2012.

19 G.R. Johnson and W.H. Cook. Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures. *Engineering Fracture MEchanics*, 21(1):31–48, 1985.

20 D.R. Lesuer, G.J. Kay, and M.M. LeBlanc. Modeling large-strain, high-rate deformation in metals. In *Third Biennial Tri-Laboratory Engineering Conference Modeling and Simulation, Pleasanton, CA, November 3-5, 1999*, 2001.

21 B. Li, X. Jiang, J. Yang, and S. Liang. Effects of depth of cut on the redistribution of residual stress and distortion during the milling of thin-walled part. *Journal of Materials Processing Technology*, 216:223–233, 2015. doi:10.1016/j.jmatprotec.2014.09.016.

22 J.G. Li and S.Q. Wang. Distortion caused by residual stresses in machining aeronautical aluminum alloy parts: recent advances. *International Journal of Advanced Manufacturing Technolgy*, 89:997–1012, 2016. doi:10.1007/s00170-016-9066-6.

23 C.R. Liu and Y.B. Guo. Finite element analysis of the effect of sequential cuts and tool-chip friction on residual stresses in a machined layer. *International Journal of Mechanical Sciences*, 42:1069–1086, 2000. doi:10.1016/S0020-7403(99)00042-9.

24 Y. Ma, P.F. Feng, and D.W. Yu. Finite element analysis ofresidual stresses and thin plate distortion after face milling. In *Proceedings of 2015 12th International Bhurban Conference on Applied Sciences and Technology, Islamabad, Pakistan*, pages 67–71, 2015. doi:10.1109/IBCAST.2015.7058481.

25 A. Madariaga, I. Perez, P.J. Arrazola, R. Sanchez, J.J. Ruiz, and F.J. Rubio. Reduction of distortions in large aluminium parts by controlling machining-induced residual stresses. *International Journal of Advanced Manufacturing Technolgy*, 97:967–978, 2018. doi:10.1007/s00170-018-1965-2.

26 T.D. Marusich and M. Ortiz. Modelling and simulation of high-speed machining. *International Journal for Numerical Methods in Engineering*, 38:3675–3694, 1995. doi:10.1002/nme.1620382108.

27 J.K. Rai and P. Xirouchakis. Finite element method based machining simulation environment for analyzing part errors induced during milling of thin-walled components. *International Journal of Machine Tools and Manufacture*, 48:629–643, 2008. doi:10.1016/j.ijmachtools.2007.11.004.

28 V. Richter-Trummer, D. Koch, A. Witte, J.F. dos Sanos, and P.M.S.T. de Castro. Methodology for prediction of distortion of workpieces manufactured by high speed machining based on an accurate through-the-thickness residual stress determination. *International Journal of Advanced Manufacturing Technolgy*, 68:2271–2281, 2013. doi:10.1007/s00170-013-4828-x.

29 C. Shet and X. Deng. Residual stresses and strains in orthogonal metal cutting. *International Journal of Machine Tools and Manufacture*, 43:573–587, 2003. doi:10.1016/S0890-6955(03)00018-X.

30 W.M. Sim. Challenges of residual stress and part distortion in the civil airframe industry. *International Journal Microstructure and Material Properties*, 5:446–455, 2010. doi:10.1504/IJMMP.2010.037621.

31 Dassault Systems. Simulia ABAQUS 6.14 Abaqus/CAE User's Guide, 2014.

32 Dassault Systems. Simulia ABAQUS 6.14 Analysis User's Guide Volume III: Materials, 2014.

33 Z.T. Tang, T. Yu, L.Q. Xu, and Z.Q. Liu. Machining deformation prediction for frame componentsconsidering multifactor coupling effects. *International Journal of Advanced Manufacturing Technolgy*, 68:187–196, 2013. doi:10.1007/s00170-012-4718-7.

34 M. Wan, X. Ye Y. Yang, and W. Zhang. Theoretical prediction of machining-induced residual stresses in three-dimensional oblique milling processes. *International Journal of Mechanical Sciences*, 133:426–437, 2017. doi:10.1016/j.ijmecsci.2017.09.005.

35 F.Z. Wang, J. Sun, J. Zhou, and J.F. Li. Experiments and Finite Element Simulations on Micro-Milling of Aluminium Alloy 7050-T7451. *Advanced Material Research*, 500:556–562, 2012. doi:10.4028/www.scientific.net/AMR.500.556.

36 J. Wang, D. Zhang, B. Wu, and M. Luo. Numerical and empirical modelling of machining-induced residual stresses in ball end milling of Inconel 718. *Procedia CIRP 58 - Proceedings of the 16th CIRP Conference on Modelling of Machining Operations*, pages 7–12, 2017. doi:10.1016/j.procir.2017.03.177.

37 D. Weber, B. Kirsch, C.R. D'Elia, B.S. Linke, M.R. Hill, and J.C. Aurich. Concept to analyze residual stresses in milled thin walled monolithic aluminum components and their effect on part distortion. In *Production at the leading edge of technology - Proceedings of the 9th Congress of the German Academic Association for Production Technology*, pages 287–296, 2019. doi:10.1007/978-3-662-60417-5_29.

38 T. Özel and E. Zeren. Finite Element Method Simulation of Machining of AISI 1045 Steel With A Round Edge Cutting Tool . In *Proceedings of the 8th CIRP International Workshop on Modeling of Machining Operations, Chemnitz, Germany*, pages 533–542, 2005.

Modeling of Nanoindentation in Ni-Graphene Nanocomposites: A Molecular Dynamics Sensitivity Study

Vardan Hoviki Vardanyan
Physics Department and Research Center OPTIMAS, Technische Universität Kaiserslautern, Germany

Herbert M. Urbassek
Physics Department and Research Center OPTIMAS, Technische Universität Kaiserslautern, Germany

Abstract

Using molecular dynamics simulation, we perform nanoindentation simulations on a Ni-graphene model system, in which a graphene flake coats the grain boundary of a Ni bi-crystal. Material strengthening or weakening by inclusion of graphene is discussed with the help of the force needed to indent to a specified depth. By varying the depth of the graphene flake with respect to the indentation depth we identify the distance up to which graphene influences the indentation behavior. In addition, we vary the details of the modeling of the graphene flake in the matrix metal and determine their influence on the performance of the nanocomposite. Our results indicate that the modeling results are robust against variations in the modeling of the graphene flake.

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases molecular dynamics, nickel-graphene composites, dislocations, nanoindentation

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.12

Funding We acknowledge the financial support of the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project number 252408385 – IRTG 2057.

Acknowledgements Simulations were performed at the High Performance Cluster Elwetritsch (RHRK, TU Kaiserslautern, Germany).

1 Introduction

Nanoindentation is a well-established tool for studying the strength of materials [4, 1]. Molecular dynamics simulation has been frequently employed to investigate the processes occurring under nanoindentation with atomistic detail [21, 20]. In ductile materials, it allows in particular to monitor the generation of dislocations during the indentation and their subsequent propagation into the material as well as dislocation reactions, annihilations or interactions with interfaces.

Recently, nanocomposite materials – and here in particular graphene-metal nanocomposites – came into the focus of research [5]. This class of materials uses graphene with its high in-plane elastic modulus and yield strength as a strengthening filler component in a metal matrix [19, 32, 28, 30]. For the understanding of plastic processes in this composite material, the interaction of dislocations with graphene flakes is relevant. Indeed, a number of simulation studies investigated this interaction. They found that as a rule graphene blocks the penetration of dislocations, thus increasing the density of dislocations near the indenter [14, 12, 3, 22]. However, the graphene interface may also absorb dislocations which eventually results in a wrinkling of the graphene layer [25].

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 12; pp. 12:1–12:13

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

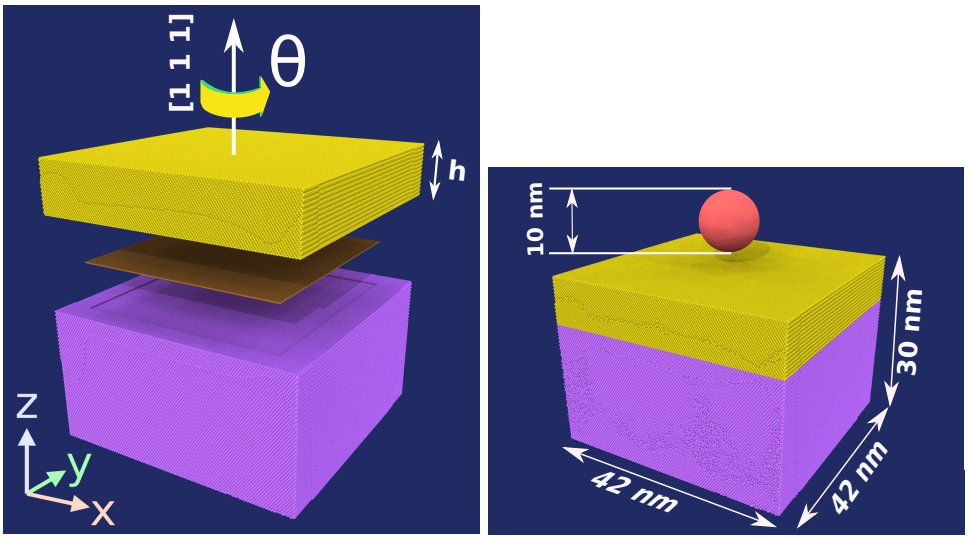


Figure 1 Left: Schematic setup of the bi-crystal used. The lower (purple) and the upper (yellow) Ni block are separated by a twist grain boundary with twist angle θ. In the grain boundary, a graphene layer (brown) is inserted. The Ni blocks have (111) surfaces. Right: Position of the indenter (red) at the beginning of the simulation and sizes of the simulation system.

In this paper, we will focus on Ni as matrix metal. Ni is a prominent matrix material and has been used in a variety of both experimental and computational studies of Ni-graphene nanocomposites [2, 11, 10, 15, 31, 16, 27, 25]. (111) surfaces of fcc metals have the same symmetry as graphene, and indeed often this is the preferred orientation of the metal-graphene interface [11]. However, because of the lattice mismatch between Ni and graphene, the interface is incoherent.

Recently, the hardness of Ni(100)-graphene [25] and Ni(111)-graphene [26] composites was studied in detail. As a model case, one may consider a Ni bi-crystal whose grain boundary is coated by a graphene flake. For this case, it could be concluded that the ideal Ni crystal is the hardest. Grain boundaries weaken it, in particular those with an incoherent and weak (i.e., high grain-boundary-energy) interface. The inclusion of graphene does not harden the composite, since (i) graphene is loaded perpendicular to its strong direction; (ii) it is opaque to dislocation slip, and dislocation absorption at the interface weakens the material; (iii) as soon as dislocations nucleate also in the lower Ni block, the indentation force is reduced. When the tip touches the graphene, it may result in interface failure, reducing the composite hardness. So in total, graphene addition does not strengthen the material in this model scenario.

In the present paper, we study to what extent details of the atomistic modeling affect the performance of the Ni-graphene interface in nanoindentation. To this purpose, we vary the exact positioning of the interface – both the depth of the graphene flake, its lateral position, and the orientation of the flake with respect of the Ni matrix – and study its influence on the nanoindentation. In addition, we also study how the interaction between graphene and Ni influences the results.

2 Simulation method

We use a Ni bi-crystal as simulation system, see Fig. 1. It has a (111) surface and contains a twist grain boundary. Two twist angles are studied in this work, $\theta = 60°$ and 30°. From a study of the grain boundary energy [17], is known that the 60° grain boundary is strong, while the 30° grain boundary is weak. The upper block has a height h which is taken to be 3, 5, and 8 nm. The entire Ni system has a a height of 30 nm and lateral extensions of 42 nm, containing approximately 4.9 million atoms.

A graphene flake of square shape is introduced into the grain boundary. It has a side length of 34 nm and is aligned with the lattice of the lower Ni block.

Structures containing graphene in the grain boundary will be denoted as g (graphene) systems, while elemental Ni systems will be denoted as hm (homointerface) systems.

We use the Mishin potential [13] to model the Ni-Ni interactions and the AIREBO potential [23] for the C-C interactions. The interaction between Ni and C is modeled by a Lennard-Jones potential according to Ref. [7].

The indenter is modeled non-atomistically as a spherical tip of radius 5 nm. It interacts with the substrate atoms via a repulsive potential according to the recipe of [8]. During the simulation, the indenter moves with a velocity of 20 m/s into the workpiece to a depth of 5 nm.

For each system, we perform 5 individual indentation simulations which differ from each other by the exact positioning of the indenter; it was moved in lateral direction randomly to another position by around ± 2 Å. Data shown are averages over these 5 simulations, unless specified otherwise.

In the Appendix 5, we also study how the orientation of the graphene flake with respect to the indentation direction influences the indentation process. For this purpose, graphene flakes oriented parallel to the indentation direction are introduced in the substrate.

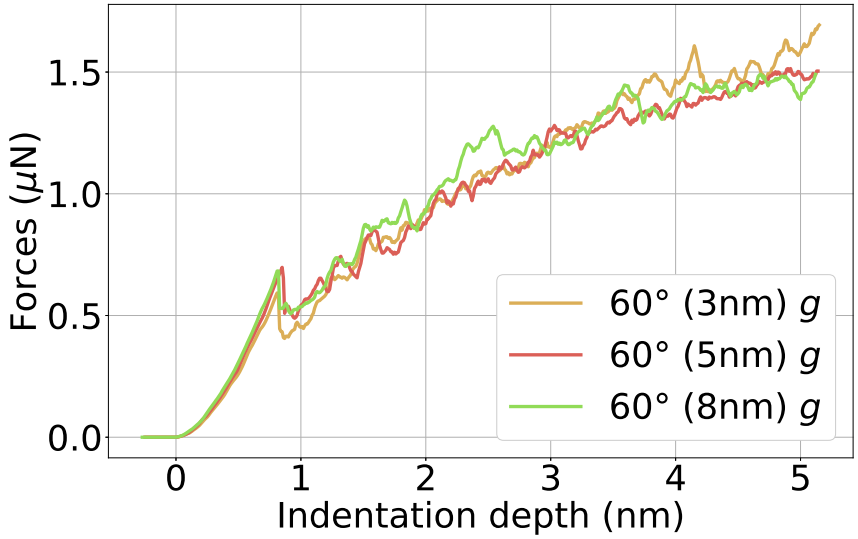
The simulations are performed with the open-source code LAMMPS [18] using a constant time step of 1 fs.

3 Results

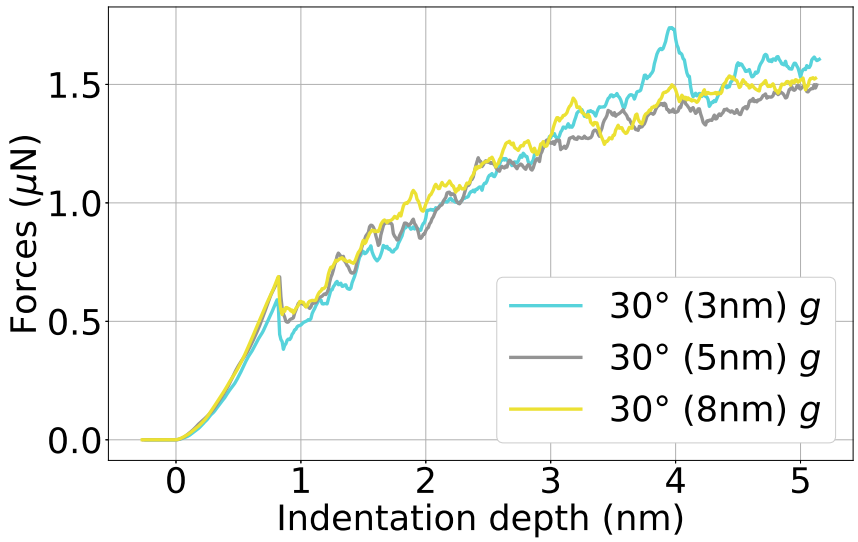
3.1 Influence of the depth of graphene

We display in Fig. 2 the forces exerted on a Ni-graphene composite during indentation to a depth of 5 nm. Three different depths of the graphene flake are compared with each other: (i) 3 nm – here the indenter touches the flake during indentation; (ii) 5 nm – here the flake is only touched at the deepest point of indentation; (iii) 8 nm – the indenter never touches the flake. Note that the figure shows averages over 5 simulations, since it is known that individual indentation events – differing by the exact indentation point – may vary from each other due to the statistical nature of dislocation generation and movement [26].

Results are shown in Fig. 2b for two different grain boundaries: for a strong grain boundary – the 60° twist grain boundary – in Fig. 2a and for a weak grain boundary – the 30° twist grain boundary. We first discuss the data for the strong grain boundary, Fig. 2a. Apart from statistical fluctuations, the data show astonishingly little difference. In particular, when the indenter touches the interface for the $h = 3$ nm flake, the curves remain unaffected. Only at depths > 3 nm, after the indenter touched the interface and starts bending down the graphene flake, the force required for indentation for the $h = 3$ nm flake is largest. The force-depth curves for the graphene flakes positioned deeper inside the Ni matrix, 5 and 8 nm, show no differences.



(a)



(b)

Figure 2 Force-depth curves for indentation in a Ni bi-crystal containing a (a) 60° and (b) 30° twist grain boundary coated with graphene. Data are shown for three different depths h (3, 5, and 8 nm) of the grain boundary.

The results are similar for the the weak 60° twist grain boundary, Fig. 2b. Again the indentation forces required for graphene flakes positioned at various depths are quite similar to each other. Only for the shallowest flake, situated a 3 nm, a strong increase in force is seen at the indentation depth of 4 nm, after the indenter touched the flake. A detailed analysis of the plastic processes inside the sample [26] shows that the peak is caused by stress increase immediately before dislocations are generated in the lower Ni block; the generation of plasticity then partially relieves the stress.

For both bi-crystal systems, a systematic influence of the graphene depth shows up at the position of the first load drop, at around 0.8 nm. This load drop is caused by the emission of dislocations due to the high stresses caused by the indenter; the creation of these defects partially releases the stress and leads to the load drop. However, the created dislocations already touch the flake in the case the shallowest flake situated at 3 nm depth; this leads to the smaller indentation force in this case.

We conclude that the exact positioning of the graphene flake beneath the indenter has only a small influence on the force of the indenter. Only when the indenter touches the graphene, and at the initial load drop, when dislocations are generated, the exact position of the flake is relevant.

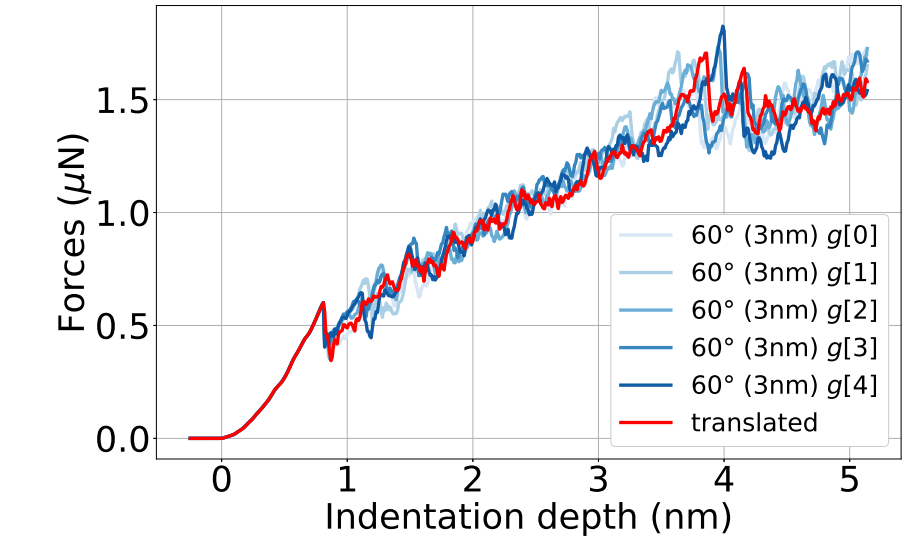
3.2 Influence of graphene size and orientation

The graphene flake used up to now had a square shape of side length 34 nm, and was aligned with the (111) surface of the lower Ni block. In the following we will focus on the Ni bi-crystal containing a 60° twist grain boundary at a depth of 3 nm, and will denote this setting as the "standard case". In Figs. 3 and 4, we vary the size and exact positioning of the graphene flake and study its consequences on the indentation force. The "standard case" simulations are included in these plots as a reference; individual results are shown in order to allow to asses their variance.

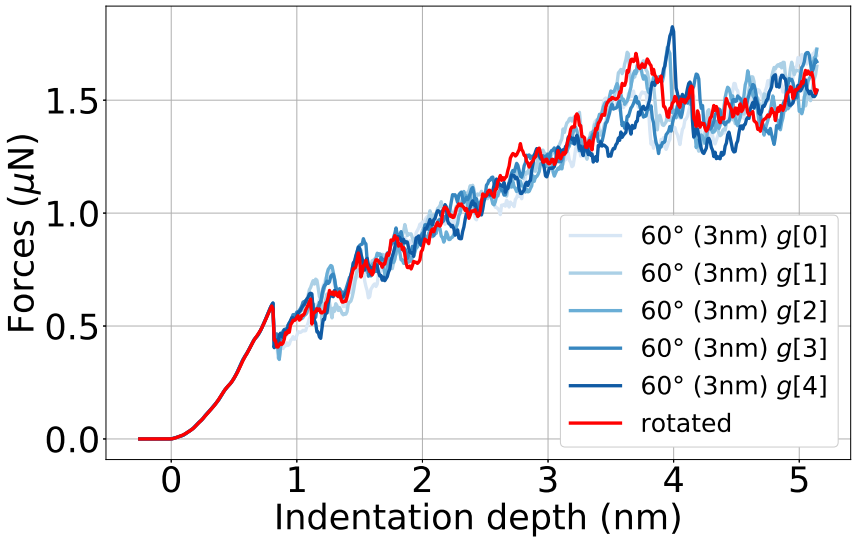
Fig. 3a gives an example of how the indentation force varies, if the lateral position of the graphene flake is slightly changed by ± 2 Å. Note that due to the lattice mismatch between graphene and the Ni(111) surface of 2.9 %, the graphene lattice is incongruent with the Ni(111) surface and so there is no "optimum" lateral positioning (on the atomic scale) which minimizes the interaction energy. However, this figure shows us that a different lateral graphene position does not change the response of the composite to indentation in any statistically relevant manner.

Fig. 3b studies a second issue, namely the orientation of the graphene flake. In the standard case, we assumed that the flake is oriented with respect to the lower Ni block. We now introduce a misorientation of 30°; this corresponds to a maximum misorientation, since for a rotation by 60°, the flake would be aligned with the upper Ni block. Again, the figure shows no statistically relevant effect on the indentation force. This may appear surprising since the graphene flake is now misaligned both with respect to the upper and the lower Ni block. However, since the interface is incoherent anyway, even a rotation of the graphene flake does not disturb the performance of the interface under indentation.

Finally, we investigate the influence of the size of the flake on the force-depth curves. On the one extreme, we expanded the flake laterally infinitely, thus mimicking a macroscopically large graphene sheet. This was achieved by extending the graphene flake to the edges of the simulation volume and using laterally periodic boundary conditions. As Fig. 4a shows, the results are statistically indistinguishable from a finite flake of side length 34 nm. Only when the indenter touches the flake, the "infinite" sheet shows a higher resistance, as demonstrated by its increased force. This is because bending of the interface – which has to occur after the indenter touched the interface, since the graphene is too strong to tear – is more difficult for the larger flake.

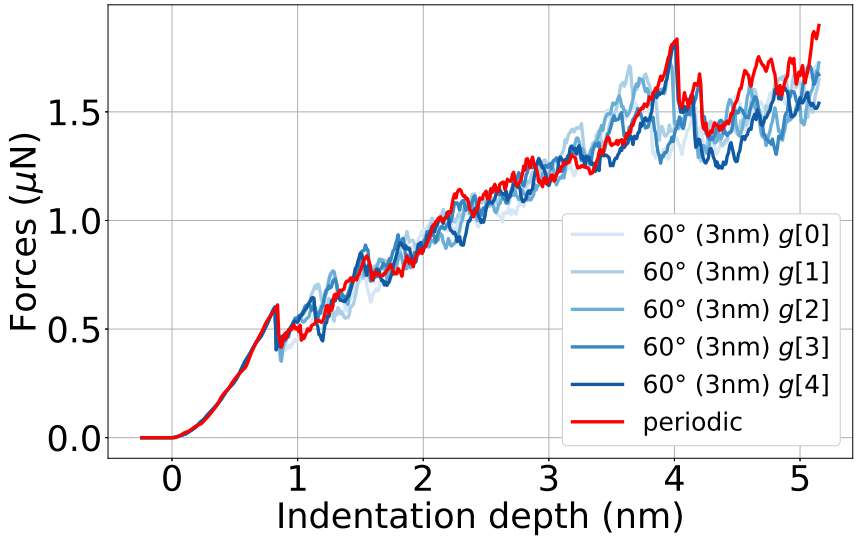


(a)

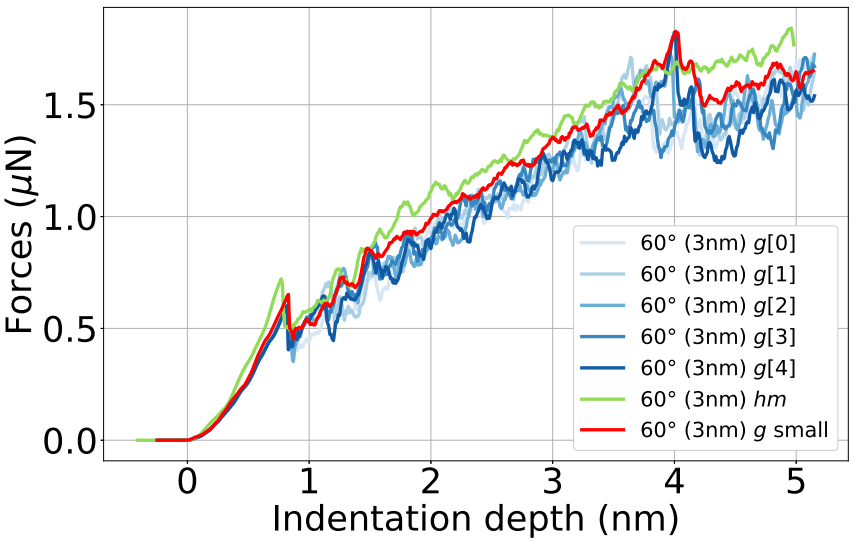


(b)

Figure 3 Force-depth curves for indentation in a Ni bi-crystal containing a 60° twist grain boundary coated with graphene. Results are shown for a flake (a) with varied lateral position, (b) rotated by 30°, (c) with laterally infinite size, as mimicked by periodic boundaries, and (d) for a smaller flake of edge length 10 nm. Simulation data are compared to individual indentation results of the standard case.



(a)



(b)

Figure 4 As in Fig. 3, but for a flake (a) with laterally infinite size, as mimicked by periodic boundaries, and (b) for a smaller flake of edge length 10 nm. Simulation data are compared either to individual indentation results of the standard case (a), or with averages of the results of standard cases (b).

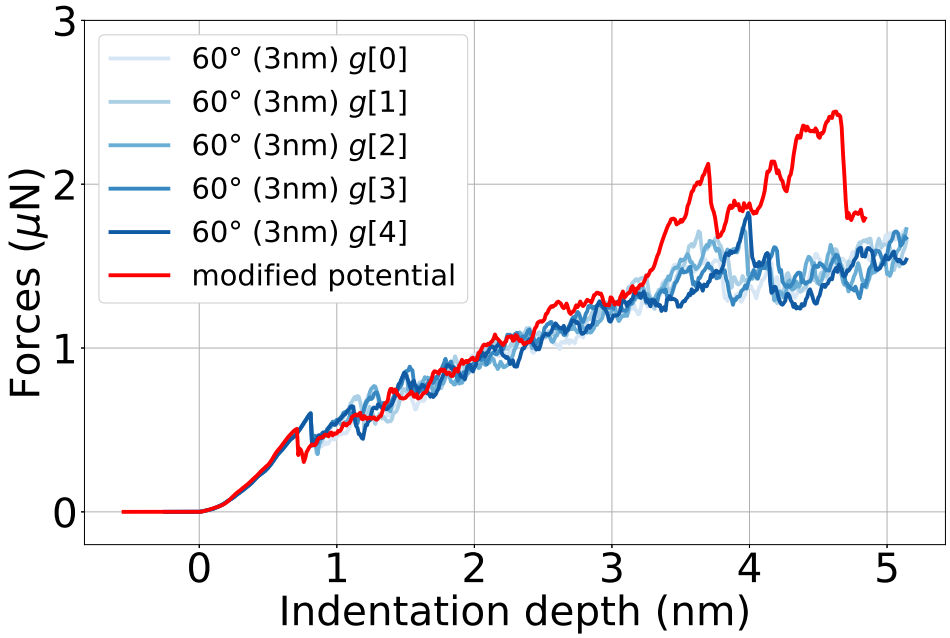


Figure 5 Force-depth curve for indentation into a 60° grain boundary at 3 nm depth coated by graphene. The results of a changed Ni-graphene potential are compared to individual indentation results for the standard case.

On the other hand, we decreased the size of the flake to only 10 nm × 10 nm. Here, the simulation (Fig. 4b) shows a hardening as compared to the standard flake size of 34 nm, but the hardening does not quite reach the values of the pure Ni sample. The general decrease of the force – compared to the pure Ni case – before the indenter touches the graphene is caused by the absorption of dislocations in the interface [25, 26]. A smaller flake can absorb only fewer dislocations such that with decreasing size of the graphene flake, the results continuously converge to those of the case of elemental Ni. After the indenter touched the interface, at around 4 nm indentation depth, we see a force maximum followed by a sharp decrease, which is reminiscent of the behavior of the 30° twist boundary, see Fig. 2b. This occurs because dislocations start nucleating in the lower Ni block and release the stress that has built up.

We conclude that the exact positioning of the graphene flake and its orientation do not alter the performance of the composite during indentation, since the quality of the incoherent interface cannot deteriorate further. On the other hand, its size matters to some extent. Smaller flakes show increased strength, since less dislocations are absorbed in the interface. Larger flakes, however, only show changes when the indenter touches the interface; they then appear harder since graphene bending is rendered more difficult. This demonstrates that our size of 34 nm was chosen well for studying the effects of large flakes under indentation.

3.3 Influence of the Ni-graphene interaction potential

The interaction between graphene and the surrounding metal is usually modeled by a simple Lennard-Jones (LJ) potential [7, 2, 10, 16, 29],

$$V(r) = 4\varepsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right]. \tag{1}$$

This simple interaction presumes that all covalent bonds within graphene are saturated such that the C atoms will only have van-der-Waals interactions with the surrounding metal. We follow here the argumentation of Huang *et al.* [7] that the C-Ni interaction is at least one order of magnitude smaller than that of the Ni-Ni interaction. This paper recommends the LJ parameters $\varepsilon = 23.049$ meV and $\sigma = 2.852$ Å.

We now study the sensitivity of our results towards changes in the potential parameters for the Ni(111) surface. Indeed, several ab-initio calculations studied the complexities of the Ni-graphene interaction. Khomyakov *et al.* [9] find that the interaction involves the formation of chemical bonds that induce hybridization between graphene p_z states and metal d states. Tavazza *et al.* [24], using DFT calculations of the interaction of a C tip to a Ni surface, calculated a LJ potential [24] with $\varepsilon = 200$ meV and a length parameter of $\sigma = 1.514$ Å. This potential actually implements a strong covalent NiC bond; the atom distance ($2^{1/6}\sigma = 1.70$ Å) is close to that of a NiC dimer, 1.62 Å. We use this potential as an extreme case to illustrate the consequences of strong bonding, even though it will be unrealistic for a Ni-graphene sheet interaction.

The results for the changed Ni-C interaction are shown in Fig. 5. We see that before the tip touches the graphene, the force is not affected by the changed potential. However, upon direct contact, the indenter force strongly rises. This occurs, since the covalent strong interaction between Ni and graphene effectively increases the stiffness of the graphene sheet and requires a stronger force of the indenter to bend it. This result is in agreement with previous results [25] for the Ni(100) surface which showed that while an increase of the bond strength does not change the indentation behavior, a decrease of the length parameter lets the indentation force increase, since then the repulsive part of the Ni-C Lennard-Jones potential can transfer larger forces.

We conclude that even strong bonds between graphene and Ni – which might occur in a situation where graphene is highly defective, since graphene edge atoms might develop covalent bonds to the metal – change the hardness of the composite only when the indenter touches the graphene. Then the strong bonding adds to the out-of-plane stiffness of graphene and thus to the hardness of the composite.

4 Conclusions

We studied nanoindentation into a Ni bi-crystal containing a graphene sheet. We obtained the following findings.

1. The depth at which the graphene flake is positioned has surprisingly little effect on the indentation force. However, when the indenter touches the flake, the indentation increases due to a stress increase caused by the difficulty to nucleate dislocations in the lower Ni block. Also at the initial load drop, when dislocations are generated, the exact position of the flake is relevant.
2. Since the Ni-graphene interface is incoherent, small changes in the exact (atomic-scale) lateral positioning of the flake, and even rotations with respect of the orientation of the Ni(111) lattice have negligible effects on the indentation behavior.
3. When graphene flakes are smaller in size, the indentation force approaches that of the elemental Ni crystal; the effect of graphene diminishes. On the other hand, if a graphene sheet with periodic boundary conditions is used – so as to approximate a flake of macroscopic extension – the indentation force increases as soon as the indenter is touched, since bending of the graphene becomes more difficult.

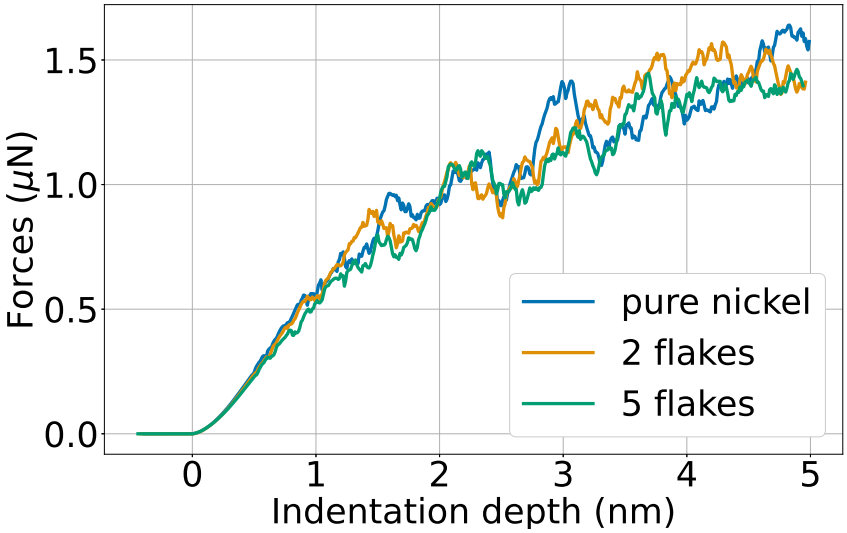


Figure 6 Force-depth curves for indentation in a Ni crystal containing vertical graphene flakes. Data are shown for three different systems: pure Ni, Ni containing 2 graphene flakes and Ni containing 5 graphene flakes.

4. In simulations, usually a pairwise interaction potential between the C atoms of the graphene flake and the Ni atoms is assumed, which models van-der-Waals-like interactions. When changing the potential to a stronger covalent-like bonding, indentation results are only affected when the indenter touches the graphene. The composite then appears harder, since bending of the graphene flake requires higher forces.

5 Vertical orientation of the graphene

In the present study as well as in previous computational work [2, 11, 31, 29, 6, 25, 26], the graphene flake is oriented parallel to the substrate surface and thus perpendicular to the indentation direction. However, it might be expected that the orientation of the graphene plane with respect to the indentation direction exerts a strong influence on the mechanical behavior. In order to study this influence, we investigate in this Appendix the extreme case, where the graphene flakes are oriented parallel to the indentation force and thus perpendicular to the surface. We simulate two scenarios: two graphene flakes placed at a distance of 20 nm positioned symmetrically such that the indenter does not touch them, and five graphene flakes at a distance of 5 nm positioned such that the indenter touches one of them at deepest penetration. The graphene flakes are embedded in single-crystalline Ni; they do not reach the surface, but end sufficiently far from the surface that the indenter touches them only at full penetration.

Fig. 6 displays the force-depth curves of these systems and compares them to the case of pure Ni. The effect of the flakes on the force evolution is indeed minor. In particular, the 2-flakes system shows a behavior that is – apart from the noise that is generated by the statistical fluctuations of dislocation generation – quite close to that of the pure-Ni system. The 5-flake system appears to be somewhat weaker than the others. This weakening is already seen quite early, at indentation depths below 1 nm; here the presence of graphene

allows for an easier nucleation of dislocations. Since dislocation generation releases the stress in the sample, the force acting on the indenter decreases. At larger indentation depths, when dislocations may glide easily along the flakes away from the indenter, the force in the 5-flake case is slightly smaller (in average) than in the other cases.

A detailed inspection of the simulation results shows that the dislocation network generated by the indentation is constrained laterally by the graphene flakes; however, the depths reached by the dislocations are similar in all cases, since dislocation propagation into the crystal interior is not blocked by graphene in this geometry. Dislocations are not repelled by the flakes but may extend to them resulting in lateral displacements of graphene atoms similar to what was described earlier in the literature [25]. In addition, the Ni dislocations can easily glide along the graphene flakes downward in the course of the indentation. Thus, while the flakes hinder the lateral expansion of the dislocations from their point of generation near the indenter, the dislocations may move freely downward and thus release the stress build-up by the indenter. This feature explains the negligible influence of the graphene flakes on the force-depth curve, Fig. 6, in this orientation.

We conclude that the vertical arrangement of graphene flakes only little influences the indentation behavior of Ni, and leads to only a slight weakening of the sample. This is caused by the fact that Ni dislocations – once they got attached to the flakes – can glide easily along them. Thus the effect of graphene on the indentation behavior in this geometry is even smaller than with the flakes oriented perpendicular to the indentation direction, where the flakes are able to block the dislocation propagation towards the material interior.

References

1 R. W. Armstrong, W. L. Elban, and S. M. Walley. Elastic, plastic, cracking aspects of the hardness of materials. *Int. J. Mod. Phys. B*, 27:1330004, 2013.

2 Shu-Wei Chang, Arun K. Nair, and Markus J. Buehler. Nanoindentation study of size effects in nickel-graphene nanocomposites. *Philosophical Magazine Letters*, 93(4):196–203, 2013. doi:10.1080/09500839.2012.759293.

3 Qing Feng, Xiaoyan Song, Hongxian Xie, Haibin Wang, Xuemei Liu, and Fuxing Yin. Deformation and plastic coordination in wc-co composite–molecular dynamics simulation of nanoindentation. *Materials & Design*, 120:193–203, 2017. doi:10.1016/j.matdes.2017.02.010.

4 A. C. Fischer-Cripps. *Nanoindentation*. Springer, New York, 2 edition, 2004.

5 Qiang Guo, Katsuyoshi Kondoh, and Seung Min Han. Nanocarbon-reinforced metal-matrix composites for structural applications. *MRS Bulletin*, 44(1):40–45, 2019. doi:10.1557/mrs.2018.321.

6 Xin He, Qingshun Bai, and Rongqi Shen. Atomistic perspective of how graphene protects metal substrate from surface damage in rough contacts. *Carbon*, 130:672–679, 2018. iron. doi:10.1016/j.carbon.2018.01.023.

7 Shi-Ping Huang, D. S. Mainardi, and P. B. Balbuena. Structure and dynamics of graphite-supported bimetallic nanoclusters. *Surface Science*, 545:163–179, 2003. doi:10.1016/j.susc.2003.08.050.

8 C. L. Kelchner, S. J. Plimpton, and J. C. Hamilton. Dislocation nucleation and defect structure during surface indentation. *Phys. Rev. B*, 58:11085–11088, 1998.

9 P. A. Khomyakov, G. Giovannetti, P. C. Rusu, G. Brocks, J. van den Brink, and P. J. Kelly. First-principles study of the interaction and charge transfer between graphene and metals. *Phys. Rev. B*, 79:195425, May 2009. doi:10.1103/PhysRevB.79.195425.

10 Youbin Kim, Jinsup Lee, Min Sun Yeom, Jae Won Shin, Hyungjun Kim, Yi Cui, Jeffrey W. Kysar, James Hone, Yousung Jung, Seokwoo Jeon, and Seung Min Han. Strengthening effect of single-atomic-layer graphene in metal-graphene nanolayered composites. *Nat. Commun.*, 4:2114, 2013. doi:10.1038/ncomms3114.

11 Da Kuang, Liye Xu, Lei Liu, Wenbin Hu, and Yating Wu. Graphene-nickel composites. *Applied Surface Science*, 273:484–490, 2013. doi:10.1016/j.apsusc.2013.02.066.

12 XiaoYi Liu, FengChao Wang, WenQiang Wang, and HengAn Wu. Interfacial strengthening and self-healing effect in graphene-copper nanolayered composites under shear deformation. *Carbon*, 107:680–688, 2016. doi:10.1016/j.carbon.2016.06.071.

13 Y. Mishin, D. Farkas, M. J. Mehl, and D. A. Papaconstantopoulos. Interatomic potentials for monoatomic metals from experimental data and ab initio calculations. *Phys. Rev. B*, 59:3393, 1999.

14 A Misra, J P Hirth, and R G Hoagland. Length-scale-dependent deformation mechanisms in incoherent metallic multilayered composites. *Acta Mater.*, 53(18):4817–4824, 2005. doi:10.1016/j.actamat.2005.06.025.

15 S. E. Muller and A. K. Nair. Dislocation nucleation in nickel-graphene nanocomposites under mode i loading. *JOM*, 68:1–6, 2016. doi:10.1007/s11837-016-1941-y.

16 Scott E. Muller, Raghuram R. Santhapuram, and Arun K. Nair. Failure mechanisms in pre-cracked ni-graphene nanocomposites. *Computational Materials Science*, 152:341–350, 2018. doi:10.1016/j.commatsci.2018.06.013.

17 David L. Olmsted, Stephen M. Foiles, and Elizabeth A. Holm. Survey of computed grain boundary properties in face-centered cubic metals: I. grain boundary energy. *Acta Materialia*, 57(13):3694–3703, 2009. doi:10.1016/j.actamat.2009.04.007.

18 St. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *J. Comput. Phys.*, 117:1–19, 1995. *http://lammps.sandia.gov/*.

19 T. Ramanathan, A. A. Abdala, S. Stankovich, D. A. Dikin, M. Herrera-Alonso, R. D. Piner, D. H. Adamson, H. C. Schniepp, X. Chen, R. S. Ruoff, S. T. Nguyen, I. A. Aksay, R. K. Prud'Homme, and L. C. Brinson. Functionalized graphene sheets for polymer nanocomposites. *Nature nanotechnology*, 3(6):327, May 2008. doi:10.1038/nnano.2008.96.

20 Carlos J. Ruestes, Iyad Alabd Alhafez, and Herbert M. Urbassek. Atomistic studies of nanoindentation – a review of recent advances. *Crystals*, 7:293, 2017. doi:10.3390/cryst7100293.

21 Carlos J Ruestes, Eduardo M Bringa, Yu Gao, and Herbert M Urbassek. Molecular dynamics modeling of nanoindentation. In Atul Tiwari and Sridhar Natarajan, editors, *Applied Nanoindentation in Advanced Materials*, chapter 14, pages 313–345. Wiley, Chichester, UK, 2017. doi:10.1002/9781119084501.ch14.

22 Fei Shuang and Katerina E. Aifantis. Relating the strength of graphene/metal composites to the graphene orientation and position. *Scripta Materialia*, 181:70–75, 2020. doi:10.1016/j.scriptamat.2020.02.014.

23 S. J. Stuart, A. B. Tutein, and J. A. Harrison. A reactive potential for hydrocarbons with intermolecular interactions. *J. Chem. Phys.*, 112:6472–6486, 2000.

24 Francesca Tavazza, Thomas P Senftle, Chenyu Zou, Chandler A Becker, and Adri C. T. van Duin. Molecular dynamics investigation of the effects of tip-substrate interactions during nanoindentation. *J. Phys. Chem. C*, 119:13580–13589, 2015.

25 Vardan Hoviki Vardanyan and Herbert M. Urbassek. Dislocation interactions during nanoindentation of nickel-graphene nanocomposites. *Computational Materials Science*, 170:109158, 2019. doi:10.1016/j.commatsci.2019.109158.

26 Vardan Hoviki Vardanyan and Herbert M. Urbassek. Strength of graphene-coated ni bicrystals: A molecular dynamics nano-indentation study. *Materials*, 13(7):1683, 2020. doi:10.3390/ma13071683.

27 Shayuan Weng, Huiming Ning, Tao Fu, Ning Hu, Yinbo Zhao, Cheng Huang, and Xianghe Peng. Molecular dynamics study of strengthening mechanism of nanolaminated graphene/cu composites under compression. *Scientific Reports*, 8(1):3089, 2018.

28 Ding-Bang Xiong, Mu Cao, Qiang Guo, Zhanqiu Tan, Genlian Fan, Zhiqiang Li, and Di Zhang. Graphene-and-copper artificial nacre fabricated by a preform impregnation process: bioinspired strategy for strengthening-toughening of metal matrix composite. *ACS Nano*, 9(7):6934–6943, 2015. doi:10.1021/acsnano.5b01067.

29 Yuping Yan, Shangru Zhou, and Sheng Liu. Atomistic simulation on nanomechanical response of indented graphene/nickel system. *Computational Materials Science*, 130:16–20, 2017. doi:10.1016/j.commatsci.2016.12.028.

30 Zhenyu Yang, Dandan Wang, Zixing Lu, and Wenjun Hu. Atomistic simulation on the plastic deformation and fracture of bio-inspired graphene/ni nanocomposites. *Applied Physics Letters*, 109(19):191909, 2016. doi:10.1063/1.4967793.

31 Fatemeh Yazdandoost, Ayoub Yari Boroujeni, and Reza Mirzaeifar. Nanocrystalline nickel-graphene nanoplatelets composite: Superior mechanical properties and mechanics of properties enhancement at the atomistic level. *Phys. Rev. Materials*, 1:076001, December 2017. doi:10.1103/PhysRevMaterials.1.076001.

32 Peng Zhang, Lulu Ma, Feifei Fan, Zhi Zeng, Cheng Peng, Phillip E. Loya, Zheng Liu, Yongji Gong, Jiangnan Zhang, Xingxiang Zhang, Pulickel M. Ajayan, Ting Zhu, and Jun Lou. Fracture toughness of graphene. *Nature Communications*, 5:3782 EP–, April 2014.

An Improved Particle Finite Element Method for the Simulation of Machining Processes

Xialong Ye[1]
Applied Mechanics, Technische Universität Kaiserslautern, Germany

Juan Manuel Rodríguez Prieto
Mechanical Engineering Department, Universidad EAFIT, Medellín, Colombia

Ralf Müller
Applied Mechanics, Technische Universität Kaiserslautern, Germany

Abstract

Machining is one of the most common and versatile manufacturing processes in industry, e.g. automotive industry and aerospace industry. But classical numerical methods such as the Finite Element Method (FEM) have difficulties to simulate it, because the material undergoes large deformations, large strain, large strain rates and high temperatures in this process. One option to simulate such kind of problems is the Particle Finite Element Method (PFEM) which combines the advantages of continuum mechanics and discrete modeling techniques. In this study we develop the PFEM further and call it the Adaptive Particle Finite Element Method (A-PFEM). Compared to the PFEM the A-PFEM enables insertion of particles and improves significantly the mesh quality along the numerical simulation. The A-PFEM improves accuracy and precision, while it decreases computing time and resolves the phenomena that take place in machining. Because metal cutting involves plastic deformation we resort to the J_2 flow theory with isotropic hardening. At last some numerical examples are presented to compare the performance of the PFEM and A-PFEM.

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases Particle Finite Element Method, Alpha Shape Method, Metal Cutting

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.13

Funding This research is funded by the Deutsche Forschungsgemeinschaft (DFG German Research Foundation) - 252408385 - IRTG 2057.

1 Instruction

Machining is a subtractive manufacturing process. Material is removed from a workpiece by a machining tool to obtain a desired shape. It is the most common and versatile manufacture processes in automotive and aerospace industry. The experimental approach to study this process is very expensive and time consuming. Therefore, numerical simulations seem to be another option. The most widely used numerical method for solving problems of engineering is the Finite Element Method (FEM) [4], which is appropriate for modelling of complex material behaviour but problematic for large configurational changes. That is because the result of the FEM is highly dependent on the quality of the mesh [15]. In machining process the material undergoes large deformations and large configurational changes. If we use a Lagrangian formulation, the mesh moves with the material and the elements become so distorted that some elements may have negative Jacobian and the numerical simulation breaks down [7]. A possible alternative is using meshless methods such as molecular dynamics (MD) and material point method (MPM), which are suitable for simulation of problems with large configurational changes, but the main disadvantage is that they are computationally

[1] corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 13; pp. 13:1–13:9
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

expensive for large length and time scales. The Particle Finite Element Method (PFEM) which was first applied for problems with liquid solid interaction combines the benefits of the FEM and the meshless methods [5]. During the simulation by using the PFEM the boundary is repeatedly detected by the α–shape method, which is appropriate for the simulation of cutting process. The domain is continuously remeshed, which make simulations with large configurational changes possible [11]. But PFEM still has some difficulties in machining simulations. In most PFEM simulations the number of the particles is fixed along the simulation, that gives rise to dense distribution of particles in some regions meanwhile sparse distribution of particles in other regions. The reported situation generates low quality of the finite elements and in some cases spurious holes. To improve the PFEM an Adaptive Particle Finite Element Method (A-PFEM) is introduced, which enables insertion of particles where the elements become too large during the simulation. Moreover a node to surface strategy is applied to solve the contact problem in this work, so near the tool tip some elements may overlap with the tool. This kind of elements should be removed.

The outline of this work is as follows. In section 2 an overview of the PFEM is illustrated. An improved PFEM called A-PFEM is introduced. In section 3 the constitutive model used in this work is explained. In the last section a couple of numerical simulations are demonstrated to show the efficiency of the proposed method.

2 Particle finite element method (PFEM)

The PFEM initially implemented for fluid mechanics problems is also a powerful tool for simulation of solid mechanics [12]. For PFEM an updated Lagrangian approach is applied to track the motion of the particles and a remeshing technique is used to overcome the problem of mesh distortion [8]. An overview of the PFEM is illustrated in Fig. 1. The computational

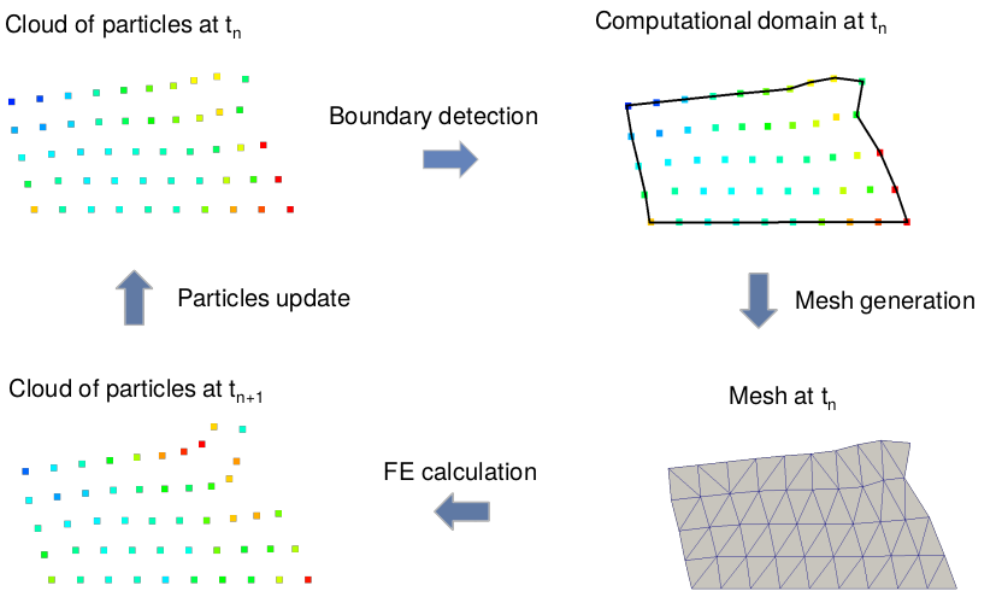


Figure 1 Operations in a standard PFEM algorithm.

domain is represented by a cloud of particles, which carry the physical quantities such as the deformation gradient, plastic deformation gradient and hardening variables. Different color of particles means different value of the physical quantities. The boundary of the particles is identified by using the so called α-shape method [2], that we will elaborate in the following subsection. For the FEM calculation we still need a finite element mesh. In this work we use the program triangle [13] to mesh the domain and the finite element programme FEAP [14] to solve the finite element problem. After that we update the particle position by using the result of the displacement obtain from the FEM calculation and project the physical quantities from the Gauß points to the particles for the next time step.

2.1 Alpha shape method

One of the key technique in the PFEM is the α-shape method, which stems from the field of computational geometry [3]. There are two versions of the α-shape method. We name them α-shape method 1 and 2 respectively for the following. For α-shape method 1 let S be the set of the particles defining the domain and h_{min} is the smallest distance between any two particles in S. For every pair of particles in S there may be two circles with radius αh_{min} that intersect both particles. If there is at least one of the both circles empty, i.e. no other particles are located within the circle, the two particles form a boundary segment. Here α is the crucial parameter which controls the shape of the detected boundary. For a uniform distribution of particles the value of α between 1.1 and 1.5 provides a good estimation of the boundary [12]. An example of this kind of α-shape method is illustrated in Fig. 2. The

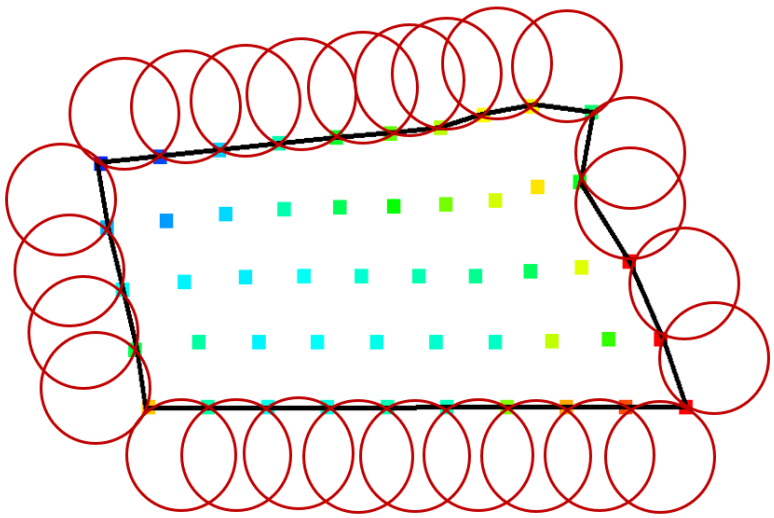


Figure 2 α-circles and the detected boundary.

problem of the α-shape method 1 is that the particles cannot keep uniform distribution along the machining process. Some particles move to each other, so that h_{min} tends to be zero. Therefore if the above α-shape method is adopted in the PFEM, the particles should be uniform distributed at the initial state and the h_{min} is only calculated once at the first time step and keep it invariant during the following simulation [13]. This version of the α-shape method may work in the PFEM, but the smallest distance h_{min} lost its meaning after the first time step.

Whereas in the α-shape method 2 the detected shape depends locally only the particles surrounding it [10]. That means for this version the particles don't need to be uniform distributed. For that we use the Delaunay triangulation to mesh the particles first and remove the unnecessary triangles for obtaining the shape of the particles. This can be achieved as following. For every element e let h_e be the smallest distance of two particles in element e and r_e be the radius of the circumcircle of the element e. For every node n let h_n be the mean value of h_e for all element surrounding this node. The distortion index α_e of the element e can be defined as

$$\alpha_e = \frac{r_e}{h_{ave}}, \tag{1}$$

where h_{ave} is the mean value of h_n for element e. For all elements if $\alpha_e > \alpha$ it will be removed, here α is the crucial parameter. An example of the α-shape method 2 is shown in Fig. 3.

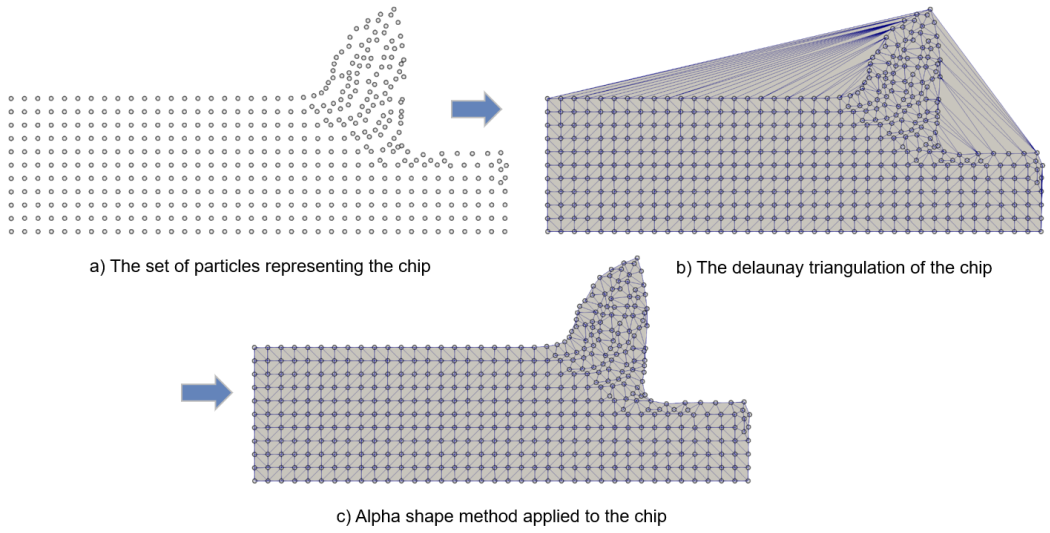


Figure 3 The steps in α-shape method 2.

2.2 Adaptive particle finite element method (A-PFEM)

The PFEM is a powerful tool thanks to the repetitive boundary detection and remeshing procedure, but still has difficulties in the numerical simulation of metal cutting process. It is impossible to avoid the presence of poor quality elements, that means the elements have small aspect ratios, and in some cases artificial holes appear as shown in Fig. 4. That is because keeping the same number of particles along the numerical simulation results in a nonuniform distribution of particles. In some regions the distribution becomes very dense meanwhile in other regions it becomes very sparse.

To overcome this difficulty the Gauß point can be inserted as a new particle in elements which become too large, that means if the area of a element is larger than μ times the initial area of this element a Gauß point will be inserted. In this work we set $\mu = 1.03$. The information of the new inserted particles can be approximated by interpolation of the

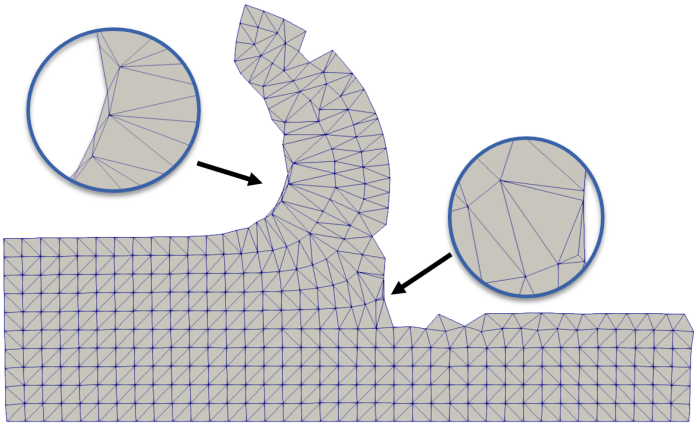


Figure 4 Artificial hole and low quality elements from a standard PFEM simulation.

surrounding particles. For simplicity the internal variables for the new particles are the internal variables available at the old Gauß points from solving the FEM problem in the last time step.

To solve the contact problem we use a node to surface strategy. So we shall especially pay attention to the nodes near the tool tip. It may occur that an element overlaps with the tool. For that we check every element. If 2 nodes of a element locate on the boundary of the tool and its centroid locates inside the tool, this element will be removed. In summary, comparing to the PFEM the A-PFEM removes bad elements after the mesh generation and inserts adaptively new points after the FE calculation in every time step.

3 The constitutive model

Nonlinear plasticity plays a key role in metal cutting. In this work we introduce a phenomenological plasticity model namely J_2 flow theory with isotropic hardening [6]. Furthermore, it is assumed that plastic flow is isochoric in accordance with the standard assumption in metal plasticity. For plasticity a multiplicative decomposition of the deformation gradient

$$\mathbf{F} = \mathbf{F}^e\mathbf{F}^p \tag{2}$$

was first introduced in [9], where $\mathbf{F}^e$ and $\mathbf{F}^p$ are the elastic and the plastic part, respectively. A stored-energy function is given by

$$W(J^e, \bar{\mathbf{b}}^e) = U(J^e) + \bar{W}(\bar{\mathbf{b}}^e) \tag{3}$$

with the volumetric part $U(J^e)$ and the deviatoric part $\bar{W}(\bar{\mathbf{b}}^e)$ defined by

$$U(J^e) = \frac{\kappa}{2}[\frac{1}{2}(J^{e2} - 1) - \ln J^e] \qquad \text{and} \quad \bar{W}(\bar{\mathbf{b}}^e) = \frac{1}{2}\mu(\mathrm{tr}[\bar{\mathbf{b}}^e] - 3), \tag{4}$$

where $J^e = \det[\mathbf{F}^e]$, $\bar{\mathbf{b}}^e = J^{e-\frac{2}{3}}\mathbf{F}^e\mathbf{F}^{eT}$. Here κ and μ are shear modulus and the bulk modulus, respectively. From Eq. (3) the Kirchhoff stress tensor $\boldsymbol{\tau}$ is derived as

$$\boldsymbol{\tau} = J^e p\mathbf{1} + \mathbf{s}, \tag{5}$$

with the Kirchhoff pressure p and the deviatoric component of the Kirchhoff stress tensor $\mathbf{s}$ defined as

$$p = U'(J^e) = \frac{\kappa}{2}(J^e - \frac{1}{J^e}) \qquad \text{and} \quad \mathbf{s} = \text{dev}[\boldsymbol{\tau}] = \mu \text{dev}[\bar{\mathbf{b}}^e]. \tag{6}$$

The yield condition, the associate flow rule and the isotropic hardening law, respectively, take the form

$$f(\boldsymbol{\tau}, \alpha) = \|\text{dev}[\boldsymbol{\tau}]\| - \sqrt{\frac{2}{3}}[\sigma_Y + K\alpha], \tag{7}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\{\bar{\mathbf{C}}^{p-1}\} = -\frac{2}{3}\gamma \ \text{tr}[\mathbf{b}^e]\mathbf{F}^{-1}\mathbf{n}\mathbf{F}^{-T}, \tag{8}$$

$$\dot{\alpha} = \sqrt{\frac{2}{3}}\gamma, \tag{9}$$

where $\mathbf{n} := \frac{\mathbf{s}}{\|\mathbf{s}\|}$, σ_Y denotes the initial yield stress, K the isotropic hardening modulus, α the hardening variable, $\bar{\mathbf{C}}$ the volume-preserving part of the right Cauchy-Green tensor $\mathbf{C}$, and γ the flow rate, which can be derived from the Kuhn-Tucker conditions and consistency condition:

$$\gamma \geq 0, \quad f(\boldsymbol{\tau}, \alpha) \leq 0, \quad \gamma f(\boldsymbol{\tau}, \alpha) = 0, \tag{10}$$

$$\gamma \dot{f}(\boldsymbol{\tau}, \alpha) = 0 \quad \text{if} \quad f(\boldsymbol{\tau}, \alpha) = 0. \tag{11}$$

4 Numerical examples

In this section we show the performance of numerical methods introduced in the previous sections. All the following simulations are based on a cutting process for a material with Young's modulus $E = 210$ GPa, Poissons ratio $\nu = 0.3$, initial yield stress $\sigma_Y = 450$ MPa and hardening modulus $K = 129$ MPa. The above parameters are used to mimic steel. In this cutting process the tool is fixed and the workpiece moves towards the tool [1]. For simplicity we let the cutting tool be a rigid body and there is only normal contact between tool and workpiece. A contour plot of the normal stresses σ_{xx} will be illustrated. In the FEM we use a linear 3-node triangle element.

4.1 PFEM with two different α-shape method

First we use the PFEM with α-shape method 1 and 2 respectively and only focus on the quality of the mesh. In Fig. 5 it can be observed that a "sawtooth" formed by using the α-shape method 1 and a hole formed by using α-shape method 2. The first phenomenon occurs because the radius of the α-circles is only defined for the initial state and too small for some later iterations. The second phenomenon happens because the local information around the hole is not correct. Besides it can be obviously seen in both examples that there are a lot of poor quality elements that have very small aspect ratio, inside the chip. Both methods can be made valid by inserting new particles in the appreciate positions along the simulation.

a) b)

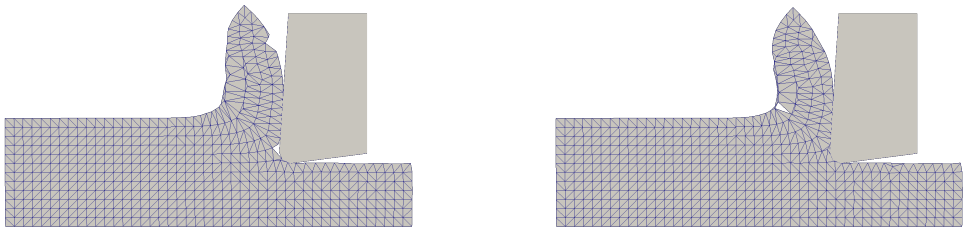


Figure 5 PFEM simulation using a) α-shape method 1 and b) α-shape method 2.

4.2 A-PFEM with two different α-shape method

We use the A-PFEM with α-shape method 1 and 2 respectively. A normal stress σ_{xx} is plotted. As illustrated in Fig. 6 high compressive stresses occur in front of the cutting tool. Comparing to the previous numerical examples both the "sawtooth" and the the hole

a) b)

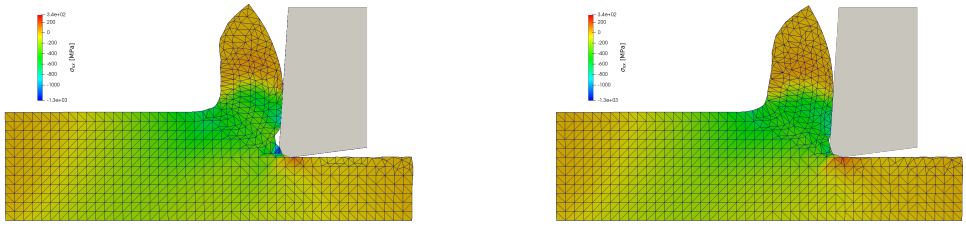


Figure 6 σ_{xx} stress using A-PFEM with a) α-shape method 1 and b) α-shape method 2.

don't appear. The elements with small aspect ratio also never show up due to the adaptive insertion of the particles. We also plot the volume of the workpiece with respect to the time step. The initial volume is 4.95. Since the workpiece has constant density, Fig. 7 shows the mass conservation for both methods.

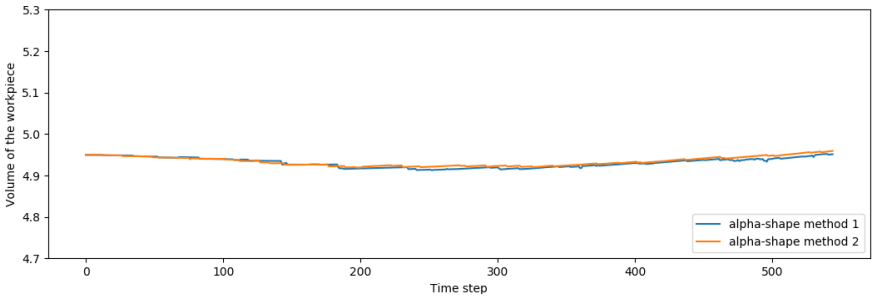


Figure 7 Workpiece volume during the A-PFEM simulation with α-shape method 1 and 2.

4.3 Comparison of PFEM and A-PFEM

We want to compare the performance of the PFEM and the A-PFEM from various aspects. For this purpose we start the cutting simulation by using the A-PFEM with much fewer particles as the simulation by using the PFEM. The normal stresses are plotted. Fig. 8 shows

a) b)

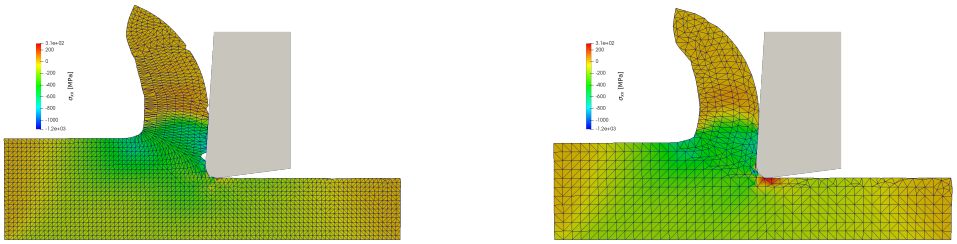


Figure 8 σ_{xx} stress using a) PFEM with refined discretisation and b) A-PFEM with coarse discretisation.

that the derived contour of both method are almost the same apart from the "sawteeth" in Fig. 8a). In Fig. 8b) it can also be observed that the inserted particles are mainly placed where the cutting process occurs and the aspect ratios of the elements in A-PFEM are obviously much smaller. Fig. 9a) shows a plot of the cutting forces of PFEM and A-PFEM. Fig. 9b) shows the computation time of PFEM and A-PFEM. Their cutting forces meet each other but the computation time by using the A-PFEM is much shorter. That is because fewer particles lead to fewer elements, and fewer elements mean a faster solution of the FEM problem, which requires most of the computation time in the simulation. In the example with A-PFEM 717 particles and 1274 elements are used comparing that 2191 particles and 4146 elements in the example with PFEM.

a) b)

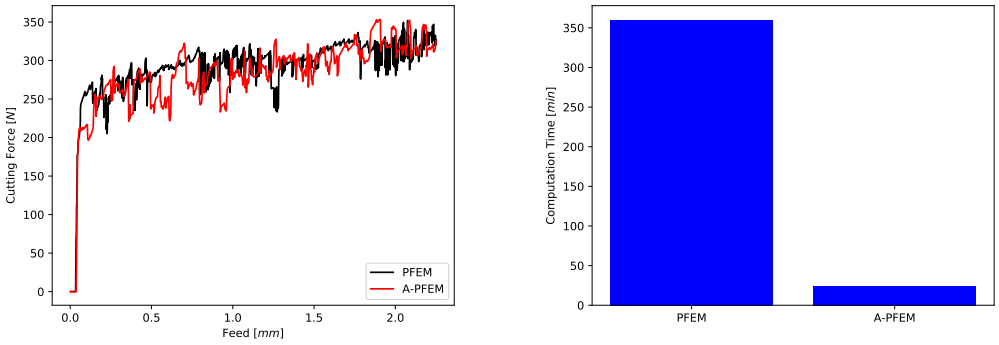


Figure 9 Comparison of a) the cutting force and b) the computation time.

5 Conclusion

An improved particle finite element method (PFEM) called adaptive particle finite element method (A-PFEM) is established in this work. In contrast to the PFEM the A-PFEM inserts new particles adaptively along the numerical simulation. In the A-PFEM the distribution of particles, which represent the computation domain, is smoother. This leads to a reduction of computing time, because we can start with fewer particles by using the A-PFEM. The new particles will be inserted in the appropriate place during the simulation, and we get the same

quality of the results as we use the PFEM with more particles. The newly developed A-PFEM is much more efficient than the PFEM and generates better meshes. Moreover, two kinds of α-shape method are discussed in this work. The α-shape method 2 is more reasonable for the simulation of the machining process, because the distribution of the particles cannot keep uniform during the simulation. In α-shape method 2 the detected shape of the particles only depends on the particles near it. For the nonlinear plasticity problem a J_2 flow theory with isotropic hardening model was introduced.

References

1 H. K. Toenshoff B. Denkena. *Spanen*. Springer-Verlag, 2011.

2 K. Fischer. Introduction to alpha shapes. *Department of Information and Computing Sciences,Faculty of Science, Utrecht University*, 2000.

3 E. P. Mücke H. Edelsbrunner. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, 13(1):43–72, 1994.

4 G. Holzapfel. *Nonlinear Solid Mechanics*. John Wiley & Sons, 2001.

5 S. R. Idelsohn, E. Oñate, F. Del Pin, and N. Calvo. The particle finite element method. an overview. *International Journal of Computational Methods*, 1(2):267–307, 2004.

6 T.J.R. Hughes J. C. Simo. *Computational Inelasticity*, volume 7. Springer Science & Business Media, 2006.

7 A. Svoboda J. M. Rodriguez, P. Jonsén. A particle finite element method for machining simulations. *VII European Congress on Computational Methods in Applied Sciences and Engineering, Crete Island, Greece, 5–10 June 05/06/2016-10/06/2016*, 1:539–553, 2016.

8 A. Svoboda J.M. Rodríguez, P. Jonsén. Simulation of metal cutting using the particle finite-element method and a physically based plasticity model. *Computational Particle Mechanics*, 4:35–51, 2017.

9 E. H. Lee. Elastic-plastic deformation at finite strains. *Journal of Applied Mechanics*, 36(1):1–6, 1969.

10 A. Frangi M. Cremonesi and U. Perego. A lagrangian finite element approach for the analysis of fluid–structure interaction problems. *INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING*, 84:610–630, 2010.

11 R. Müller M. Sabel and C. Sator. Simulation of cutting processes by the particle finite element method. *GAMM-Mitteilungen*, 40(1):51–70, 2017.

12 J. Oliver, J.C. Cante, R. Weyler, C. González, and J. Hernández. Particle finite element methods in solid mechanics problems. *Computational plasticity*, pages 87–103, 2007.

13 J. R. Shewchuk. Triangle: Engineering a 2d quality mesh generator and delaunay triangulator. *Applied Computational Geometry: Towards Geometric Engineering*, pages 203–222, 1996.

14 R. L. Taylor. Feap - a finite element analysis program: User manual. *Department of Civil and Environmental Engineering, University of California, Berkley*, 2009.

15 P. Wriggers. *Nonlinear finite element methods*. Springer Science & Business Media, 2008.

Physical Modeling of Full-Field Time-Domain Optical Coherence Tomography

Andrej Keksel[1]
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Georgis Bulun
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Matthias Eifler
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Anis Idrizovic
The Institute of Optics, University of Rochester, NY, USA

Jörg Seewig
Institute for Measurement and Sensor-Technology, Technische Universität Kaiserslautern, Germany

Abstract

In this paper, a physical model of full-field time-domain optical coherence tomography (FF-TD OCT), which focuses the requirements of measuring inner textures of flexible layered samples in industrial applications, is developed and validated by reference measurements. Both the operating principle and the overall design of a FF-TD OCT correspond to that of classical white light interferometry (WLI), commonly used for the measurement of areal micro-topographies. The presented model accounts for optical and geometrical properties of the system, multiple scattering of light in turbid media and interference of partially coherent light. Applying this model, virtual measurements are used to exemplarily investigate the extent to which the principles of classical WLI can be directly transferred to obtain layer thickness measurements by simulating the use of a simple low-cost WLI system as OCT. Results indicate that a currently existing instrument setup can **only** be used as OCT to a **very limited** extent but not in general due to its initial design as a WLI.

2012 ACM Subject Classification Computing methodologies → Modeling methodologies

Keywords and phrases Optical coherence tomography, full-field time-domain OCT, virtual measuring, optical measurement technology, physical modeling

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.14

Funding This research is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057.

1 Introduction

In face of digitalization, industrial processes are increasingly virtualized on multiple levels. This includes the field of optical metrology, with the aim of predicting measurement results and expected uncertainties. For this purpose, the description of optical measuring instruments is increasingly based on mathematical models that represent complex physical relationships in a simplified form [29]. Such description includes a virtual measuring instrument, a virtual measuring object and, above all, the interaction between these two components [29]. The model-based prediction of measurement results can be used to, for example, analyze and optimize existing instrument setups and thus better understand the underlying processes,

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 14; pp. 14:1–14:22
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

or even provide insights for the development of new types of measurement systems [23]. In addition, virtual measurement data can be used for the optimization of the cost-benefit ratio in the early stages of planning a larger measurement series by estimating instrument-specific measurement uncertainties to be expected [30].

Current literature discusses a variety of physical modeling approaches, describing different optical measuring devices; these include models for describing white light interferometers [5, 28, 37], scattering light sensors [32, 37, 38, 39], atomic force microscopes [45] or X-ray computed tomography devices [10, 36].

Alternatively to the physical modeling approach, a characterization of the instrument transmission behavior, especially of topography measuring instruments, based on measurements of known geometries (represented by material measures) is a common procedure in practice. The results of this kind of empirical reference measurements can also be used for instrument specific uncertainty analyses [11] or simplified virtual measurements [17, 18].

In the context of this paper, a physical modeling approach is applied to develop a comprehensive model of an optical measuring principle – optical coherence tomography (OCT) – which is used for the characterization of intra-tissue textures. By customizing this model, for example to represent a low-cost instrument setup, it can be virtually examined to which extent this setup can be used to obtain expressive OCT measurement results.

2 Optical Coherence Tomography

Optical coherence tomography (OCT) is an optical imaging technique that allows the internal structure of scattering tissues to be imaged to micrometer resolution. OCT is mainly used in the medical sector as it enables "optical biopsy", allowing visualization of internal tissue structures without prior sample preparation. The procedure is considered to be the optical equivalent of ultrasound imaging (sonography). By using light instead of sound, OCT is limited to a lower penetration depth, but it features higher axial and lateral resolution. The most important application of OCT is currently located in ophthalmology (eye medicine). On the other hand, general suitability of OCT for industrial applications is not as explored. Selected industrial applications of OCT published are:

- Imaging fiber tows and voids in two materials: an epoxy E-glass-reinforced composite and a vinyl-ester E-glass-reinforced composite [8].
- Measurement and evaluation of the quality of the coating layer of premium glossy photopaper [22].
- Fast industrial inspection of an optical thin film panel (similar to a LCD panel) [33].
- Three-dimensional imaging of complete polymer solar cells using a high-resolution OCT system [35].
- Investigation of confocal enhanced OCT with improved image contrast and depth resolution by imaging a highly scattering paint layer [44].

Although the non-contact, non-invasive nature of the technology and its ability to image transparent, turbid and highly scattering media make the OCT technology attractive for non-destructive testing and evaluation of manufactured parts, existing instruments and their respective models are mainly designed to operate in the context of medical and biological research.

2.1 Physical measuring principle of OCT

The measuring principle of OCT is based on reconstructing cross-sectional images of intra-tissue textures using light reflections from the interior of the tissue. While the location of sound-reflecting objects is reconstructed by measuring the echo delay time in sonography, OCT uses a classical approach from optical metrology, namely coherence scanning interferometry (CSI), better known as white light interferometry (WLI) [6, 24, 43].

In a classical Michelson interferometer, the light emitted by a light source is split into two beams by a beam splitter, which are subsequently directed to the sample on the one hand and to a reference mirror on the other hand. The light beams reflected from the sample and the reference mirror are merged again afterwards, interfering with each other based on the wave properties of light. The obtained interference pattern has a high sensitivity to the path length difference between the object beam and the reference beam. Analysis of the pattern allows conclusions about positions of reflective tissue layers to be drawn. Combining classical microscopy with interferometric objectives, the micro-topography of a measuring object can be measured using the principle of CSI/WLI. By assigning a height value to every measured point, a topography map of the sample can be reconstructed. When measuring micro-topographies, monochromatic light is not typically used, but as the name "white light interferometry" already suggests, white or low-coherent light is used. The advantage of this is that the coherence length l_c of low-coherent light, which can be calculated by the properties of its spectrum, is limited to a few micrometers, so the interference pattern, which is the carrier of the height information, only occurs in a narrow height range of about $\Delta z \approx l_c$ (see also in figure 2(b)) leading to increased evaluation precision [4, 19, 24].

The principles of low-coherent interferometry described above are applied in OCT for the measurement of intra-tissue textures. There are various OCT systems whose operating principles differ greatly from each other and therefore feature different advantages and disadvantages depending on the measurement task with regard to resolution, detection speed, noise behavior, etc. In this context, a distinction is made between time-domain OCT (TD-OCT) and spectral-domain OCT (SD-OCT). Furthermore, OCT systems can be classified according to the type of lateral scanning. A distinction is made between common point-by-point scanning (flying spot) and areal measurement (wide-field or full-field OCT) [21].

The OCT model presented in this paper is based on the combination of time-domain and full-field OCT, which corresponds to the principle of a classical white-light interferometer for surface topography measurements. After validating the presented model by controlled reference measurements, it is exemplarily applied for investigating if, or under which circumstances, a custom low-cost WLI instrument setup, which is imaged in figure 7 and described in more detail in [2, 37], is suitable for the use as OCT or even as a WLI&OCT convertible instrument in general.

3 OCT model

Current literature proposes a variety of modeling approaches for OCT. These vary from very simple (single backscattering model), over very complex but restricted (extended Huygens-Fresnel based model), to very flexible but numerically complex approaches (Monte-Carlo methods combined with physically and/or statistically based models). Corresponding references are listed and summarized below.

Single backscattering model.

- In [13] a low coherence interferometry model was used to compare broadband single-Gaussian and multi-Gaussian light sources.
- In [14] an analytical model is used to simulate different optical delays in time domain OCT.
- In [25] OCT simulations has been performed using Lab VIEW and Matlab scripts.

Extended Huygens-Fresnel based model.

- In [34] an extended Huygens-Fresnel model is presented and used to describe the performance of OCT systems in both the single and multiple scattering regimes simultaneously.

Monte-Carlo model.

- In [15] a new model of OCT, which takes multiple scattering into account, is presented, combining Monte-Carlo methods and physically/statistically based models.
- In [20] a Monte-Carlo based simulator of OCT imaging for turbid media with arbitrary spatial distributions has been developed and demonstrated.
- In [40] a Monte-Carlo method for modeling OCT measurements of a diffusely reflecting discontinuity embedded in a scattering medium is presented.
- In [42] a Monte-Carlo model of steady-state light transport in multi-layered tissues has been developed and implemented.

Since the OCT model presented here is primarily intended to model industrial measurement tasks, which can be very diverse (measuring objects with different scattering properties, flexible layer structure, several light sources, lenses and objectives to choose from, etc.), a flexible and numerically efficient model is required. For this reason, the OCT model developed by the author takes into account not only the light propagation in scattering media and (partially) coherent interference properties of light, but also specific geometrical, optical and technical properties of the underlying optical system. In the next section, each of the three main components of the OCT model (optical system in section 3.1, light propagation in turbid media in section 3.2, interference behavior in section 3.3) is considered and described separately. Subsequently, the different model components are coupled and an efficient implementation (briefly described in section 3.4) is performed in order to obtain a comprehensive and at the same time fast-computing OCT model (time frame of minutes for obtaining expressive results) on which basis virtual measurements can be performed.

3.1 Optical system

The model of the optical system includes the beam paths data within the measuring instrument on the one hand and modeling of technical properties of relevant hardware components, which include lenses installed in the system, the light source, the interferometric Michelson objective and the detector on the other hand. The fundamental setup of the interferometer to be modeled is shown in figure 1.

Lenses and beam paths. When calculating the light ray paths within the optical system, the main focus lies on the lenses installed in the measuring instrument (collimator, tube and objective lenses), which are considered to have idealized properties, also known as “thin lenses” [26]. The ray paths within the measuring instrument (from the light source to the measuring object and from the measuring object to the detector) resulting from this assumption were previously derived in [37] and will be adapted in the context of the virtual FF-TD OCT to be modeled.

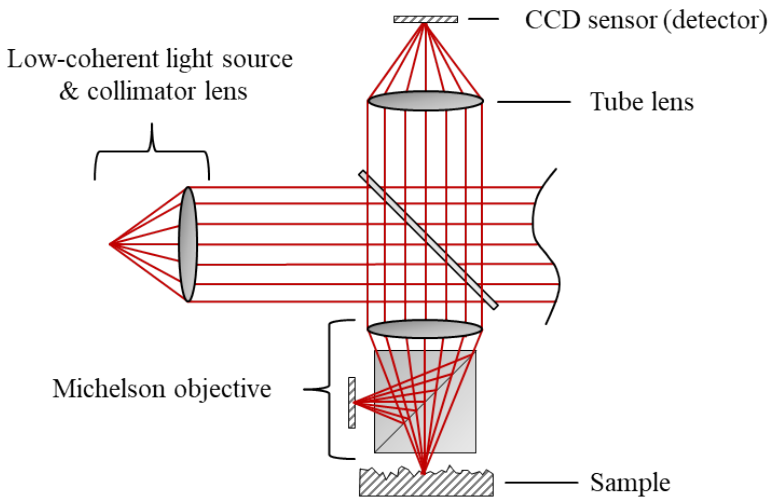


Figure 1 Schematic layout of the interferometer to be modeled.

Light source. Furthermore, the spectral properties of the light source are crucial for the function of an OCT. The electromagnetic spectrum of the light source used, which is represented by a Gaussian spectrum in figure 2 on the left, primarily describes the optical resolution limits of the measuring instrument [7]. While the lateral resolution mainly depends on the average wavelength λ_0 and the objective used (assuming the light entering the objective to be spatially incoherent [15, 16]), the axial resolution is characterized by the temporal coherence function or autocorrelation function $\gamma(\tau)$, which can be calculated according to the Wiener-Khinchin theorem by Fourier transform of the entirety of the spectrum of light used ($S(\lambda)$ in terms of wavelength λ, or $S(\nu)$ in terms of frequency ν) [6]:

$$\gamma(\tau) \sim \int_0^\infty S(\nu) \cdot \exp(-i2\pi\nu\tau) d\nu \tag{1}$$

Note that the integral only has to cover positive values since a light spectrum is exclusively composed of positive wavelengths/frequencies. In general, the axial resolving capability increases with increasing bandwidth. The real part of the autocorrelation function is shown in figure 2b and corresponds to an ideal correlogram resulting from the interference of sample and reference beam, on the basis of which measurement signals are evaluated in CSI/WLI and TD-OCT.

Another important property of light in OCT is the maximum penetration depth into the measuring sample. This also depends on the light spectrum used. In OCT measurements of biological tissue, light in the near-infrared range (wavelengths in the range of $\lambda = 650 \ldots 1350\,nm$) is commonly used, since the absorption of light in biological tissue is minimized in that wavelength range [7].

In summary, for high-resolution OCT applications, a light source with the broadest possible spectrum (high axial resolution) in the near-infrared range (high penetration depth) is recommended.

Interferometric Michelson objective. Other important components of the optical system are the interferometric objectives, which fundamentally differ in their design. The three basic designs are the Michelson, Mirau and Linnik designs [27]. In a Michelson objective

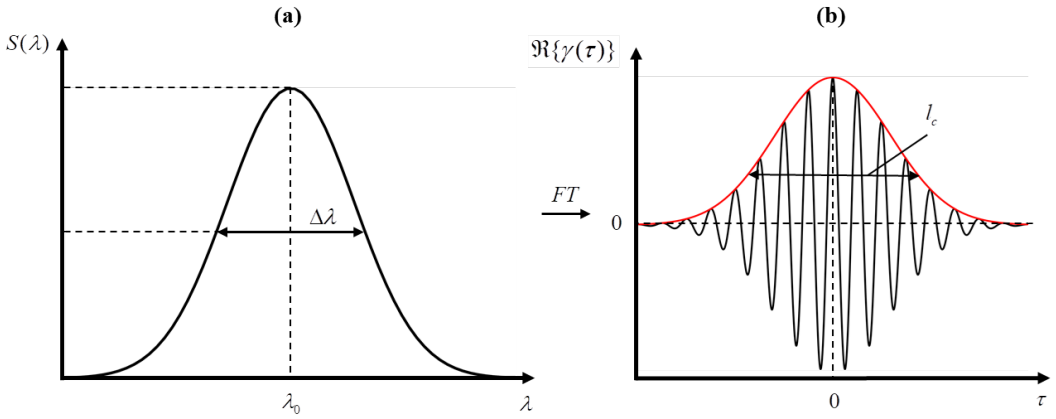


Figure 2 (a) Exemplary Gaussian light spectrum $S(\lambda)$ and (b) the real part of the corresponding autocorrelation function $\Re\{\gamma(\tau)\}$ with plotted envelope (red) and coherence length $l_c \approx \lambda_0^2/\Delta\lambda$.

(figure 3), a beam splitter is mounted below the focusing lens. This separates the incident light beam into sample and reference beam. The reference mirror is mounted on the side of the beam splitter. This design is only used for objectives with a low magnification ($\leq 10\times$), which have a larger focal length due to the limited space available and thus low numerical aperture (NA) [27]. An appropriate model of a Michelson objective is provided by following parameters: Numerical aperture (NA), beam splitting ratio of the beam splitter (ρ) and reflectance of the reference mirror (r_R). While NA represents a geometric parameter, ρ and r_R represent light intensity ratios. The beam splitting ratio ρ describes the intensity ratio between the reference beam and the sample beam, and the reflectance of the reference mirror r_R represents the ratio of the intensities of the reference beam after and before the reflection takes place.

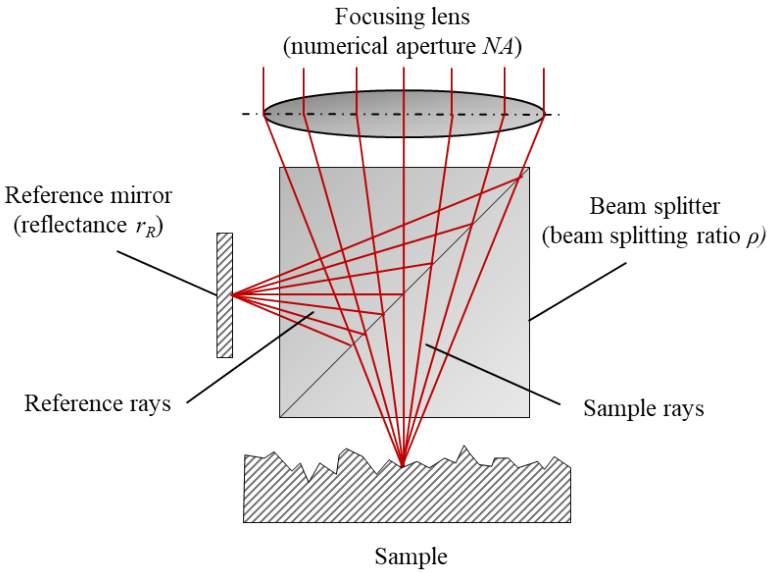


Figure 3 Michelson objective.

The modeling of the detector within this work is mainly limited to pixel size and with this the optical system to be modeled is fully characterized.

In the next section, the Monte-Carlo based modeling approach used to describe light propagation in turbid media is summarized.

3.2 Light propagation in turbid media

The light-matter interaction in the context of OCT depends both on the properties of the incident light and on the optical properties of the illuminated sample. The interplay of these properties defines the reflection, transmission, absorption and scattering behavior as well as changes in the polarization states. There are various material-specific optical parameters that describe the interaction of light and matter. These include:

- Refractive index n (reflection and transmission behavior)
- Absorption coefficient μ_a (absorption behavior)
- Scattering coefficient μ_s (scattering behavior)
- Anisotropy factor g (degree of directional change due to scattering)

It should be noted that each of these optical material properties is generally wavelength-dependent [43]. A frequently used approach for the description of light transport in multi-layered tissue using the material-specific optical parameters mentioned above is the Monte-Carlo based MCML algorithm of Wang et al. [12, 42], which is adapted here in a customized version. This approach considers light in the form of photons, or photon packages. The propagation of light is characterized by a large number of photons which are emitted by the light source and whose trajectory and intensity changes induced by reflection, transmission, scattering and absorption processes are calculated by statistical methods. The implemented customizations of the MCML algorithm are the following:

- The assumption of orthogonal and point-wise light irradiation of the tissue is replaced by the model of a focused light beam (focusing properties according to the NA of the Michelson objective used).
- The Henyey-Greenstein scattering function for the description of the material's scattering properties is replaced by the **extended** Henyey-Greenstein scattering function, also known as Cornette-Shanks phase function [3, 41], which provides a more realistic scattering pattern.
- The light rays do not experience ideal reflection or transmission when hitting an interface but are subject to roughness induced scattering, also described by the extended Henyey-Greenstein scattering function.

In figure 4, exemplary trajectories of the modeled light-matter interaction of a light beam hitting the surface of two composites perpendicularly are shown on the right. The samples are surrounded by air and are modeled as two-layered composites of glass (figure 4a) and turbid tissue (figure 4b). The corresponding optical material parameters, as well as the geometrical sample properties are summarized on the left-hand side of figure 4.

Combining the trajectories of the scattered photons within the tissue and the modeled ray paths within the optical system, the entire trajectory of a photon emitted by the light source, hitting the sample and being reflected towards the detector is fully described. At the plane of the intensity-sensitive detector, the interference of the reference and sample beam is made visible. The mathematical description of the partially coherent interference behavior at the detector plane represents the third pillar of the OCT model, which is explained in the following section.

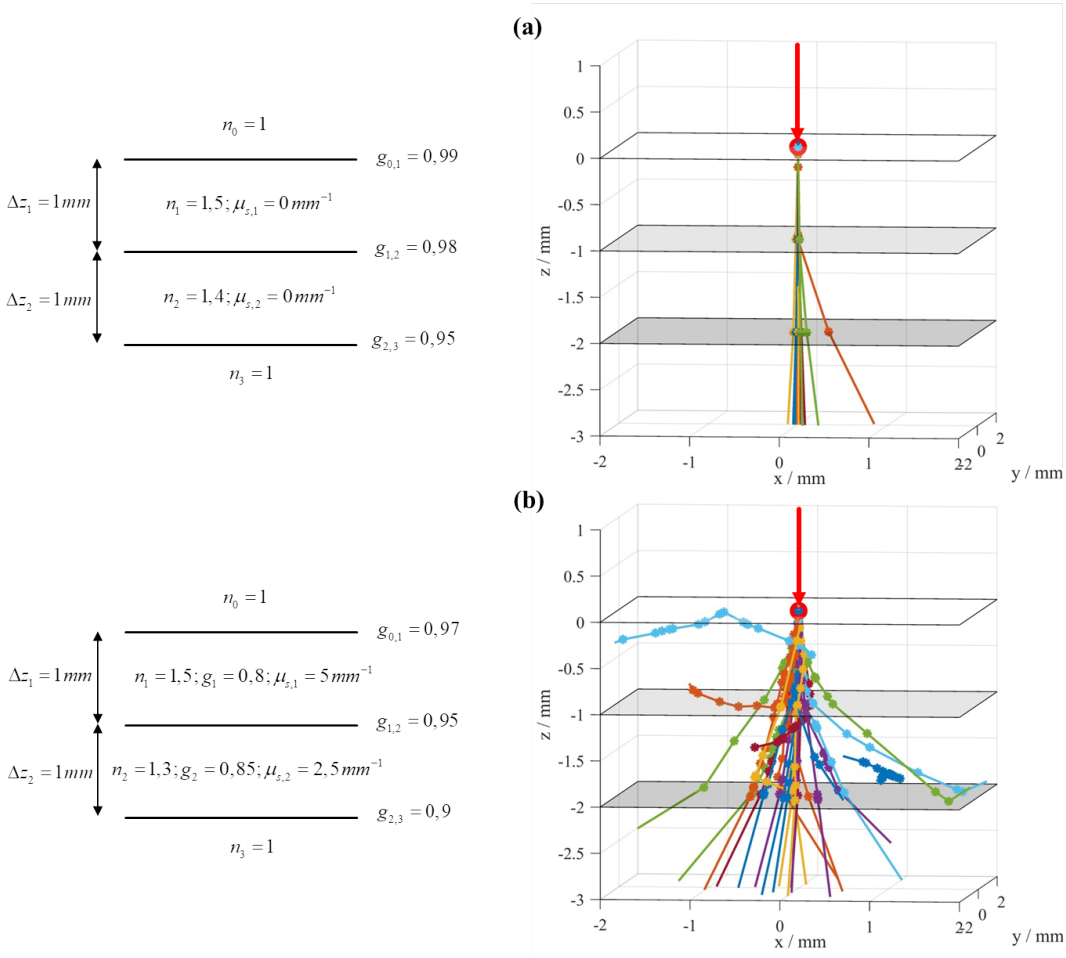


Figure 4 Geometrical properties and optical parameters of the modeled material composites (left) and 25 exemplary simulated photon trajectories resulting from the light-matter interaction of orthogonal illumination (right) of a two-layered glass composite (a) and a two-layered biological tissue (b).

3.3 Interference properties

The interference properties of the sample beam and the reference beam are modeled by classical Fourier optics [9], considering a scattered and therefore attenuated sample beam. For this purpose, both temporal and spatial coherence properties of light must be taken into account. The temporal coherence properties of a light source, key for the axial resolving capability of the measuring instrument, are entirely defined by its spectrum. This is not the case for the spatial coherence properties, which mainly influence the lateral resolution of the measuring instrument. In the context of this paper, a spatial coherence model is applied, where only two absolute modes – completely spatially coherent and completely spatially incoherent superposition of reference and sample beams – are distinguished. According to [15, 16] this assumption is valid if the light entering the objective is provided by uniformly bright spatial incoherent light, which is also a requirement for a FF-TD OCT system and thus installed in the current instrument setup. Whether reference and sample beams are coherently superimposed at the detector plane depends on their light source origin (which

must be the same for reference and sample photons), their optical path differences (which must be within the coherence length of the light to be considered as temporal coherent to each other) and their detector hitting points relative to each other (which must be within the range of the Airy disk to be considered as spatial coherent to each other). This model of coherence domain limitation corresponds to the "coherence volume" model from [15, 16]. According to this model, the Airy disk provides the lateral limitation of the coherence volume, which is, according to the Rayleigh criterion, also a measure of the achievable lateral resolution of the measuring instrument [1]. The coherence length of the light is a measure of axial limitation of the coherent volume. Based on this model, signal formation on the detector as a result of coherent as well as incoherent superposition of reference and scattered sample beams or photons is described below.

In the simplest case of ideal low-coherent two-beam interferometry, the detector signal I_D can be classically determined according to following equation [5]

$$I_D(\tau) = I_R + I_S + 2 \cdot \sqrt{I_R \cdot I_S} \cdot \Re\{\gamma(\tau)\}, \tag{2}$$

whereby I_R and I_S represent the intensities of mutually-coherent reference and sample beams, τ represents the optical path difference between the reference beam and the sample beam, and γ represents the normalized temporal coherence function (or autocorrelation function) of the light used. However, in the OCT model presented here, both the reference and sample beams are described as the sum of a large number of individual and potentially scattered photons that originate from different points of the areal light source, hit the detector on various positions with different intensities and optical path differences relative to each other and are not necessarily mutually-coherent. Therefore, the description of the detector signal must be modified accordingly. The photon-wise composition of the detector signal is visualized in figure 5. The calculation is assuming a gridded detector plane, in which each grid cell represents a coherence region of the reference beam hitting the detector. To describe the composition of the detector signal in figure 5b, the blue marked coherence region in the center of figure 5a serves as a representative coherence region of the detector.

The reference beam, or the sum of the photons forming it, is not scattered in the model. As a result, a beam ideally focused on the detector (red or black dot) assumed to be spatially incoherent creates a coherence region around itself equal to the size of the (magnification-scaled sample-plane) Airy disk, which is approximated here as an equally sized square (blue square). The intensity of the reference beam depends on the number of reference photons hitting the detector and their individual intensities. When describing the sample beam, which is also composed of a multitude of scattered photons, a distinction must be made between sample photons that are coherent and those that are incoherent to the reference beam, since coherent superposition and incoherent superposition of reference and sample beams are fundamentally different. Mathematically, the interaction of the electric/magnetic fields can be easily described by summing up the individual photon intensities in the case of incoherent superposition, which is not the case when considering the superposition to be coherent. Both the sample beam photons that are coherent to the reference beam (red crosses, originate from the same point as the reference beam represented by the red dot) and those that are incoherent to it (gray crosses, originate from other points than the reference beam represented by the red dot) hit the detector within the observed coherence region. Since sample photons experience individual scattering and absorption when interacting with the sample to be measured, their intensities and optical path differences relative to the corresponding reference path must be tracked individually and the number of photons hitting the representative region of the detector (mutually coherent and mutually incoherent) has to be counted. It

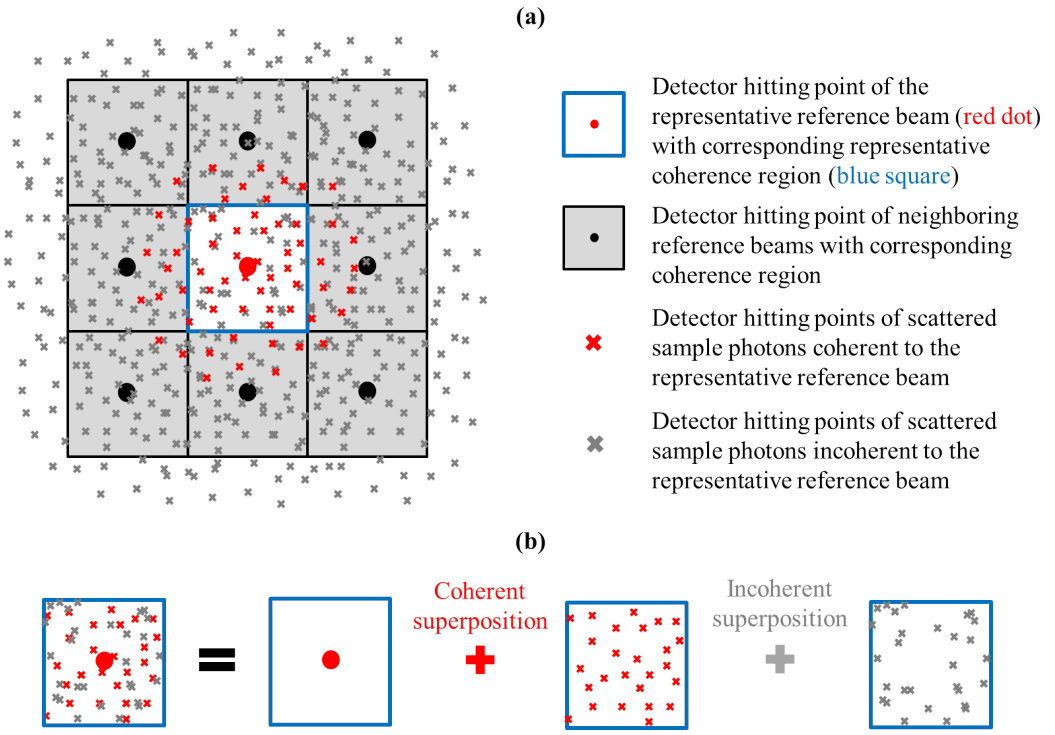


Figure 5 (a) Grid of the detector plane with visualization of the detector hitting points of exemplary reference and sample photons; (b) Composition of the detector signal consisting of reference beam, sample photons coherent and sample photons incoherent to it.

should be mentioned that only the mutually coherent sample photons (red crosses) contribute to the information-containing signal formation. The scattering-induced "cross-talk" between mutually incoherent reference and sample photon pairs (gray crosses hitting the representative blue marked region) does not provide any valuable height information, but only leads to a reduction of the signal-noise ratio by increasing the total intensity.

Based on this model of signal formation, the three summands in (2) have to be replaced by modified terms. It should be noted that only the photons hitting the representative coherence region (blue marked square) are considered forming the signal.

The intensity of the reference beam $I_{R,\Sigma}$ (red dot) depends on the number of reference photons hitting the detector $k_{R,Det}$ and their individual intensities, whereby the individual intensity of a reference photon is obtained by

$$i_R = \rho^2 \cdot r_R \cdot i_0, \tag{3}$$

considering the initial photon intensity i_0, the photon passing the beam splitter twice ($i_R \sim \rho^2$) and taking into account the reflectivity of the reference mirror ($i_R \sim r_R$). Accordingly, $I_{R,\Sigma}$ is determined by

$$I_{R,\Sigma} = k_{R,Det} \cdot i_R. \tag{4}$$

This term (4) substitutes the first summand of (2).

The situation is similar for the photons of the sample beam hitting the detector $I_{S,\Sigma}$. Sample beam photons that are both coherent (red crosses) and incoherent (gray crosses) to the reference beam hit the detector within the representative coherence region (blue square).

Since sample photons experience individual scattering and absorption when interacting with the sample to be measured, their intensity must be determined individually and the number of photons hitting the representative section of the detector must be counted. The individual intensity of a sample photon $i_{S,j}$ is obtained by

$$i_{S,j} = (1-\rho) \cdot [i_0 \cdot (1-\rho) - i_{a,j}], \tag{5}$$

considering the photon passing the beam splitter twice and experiencing an individual absorption-related intensity loss $i_{a,j}$. Accordingly, $I_{S,\Sigma}$ is determined by

$$I_{S,\Sigma}(z) = \sum_{j=1}^{k_{S,Det}(z)} i_{S,j}, \tag{6}$$

where $k_{S,Det}(z)$ is the number of sample photons hitting the representative coherence region at scan position z, regardless of whether they are coherent to the reference beam. The term in (6) substitutes the second summand of (2).

The substitution of the third summand from (2) results from the coherent superposition of reference and sample photons. In addition to the number and intensity of the mutually-coherent reference and sample photons hitting the detector at scan position z, their optical path difference to each other τ combined with the temporal coherence function $\gamma(\tau)$ of the light used affect the signal formation. This relationship is described by

$$I_{c,\Sigma}(z) = 2 \cdot \sqrt{I_{R,\Sigma} \cdot |IA_{S,c,\gamma,\Sigma}(z)|} \cdot \operatorname{sgn}(IA_{S,c,\gamma,\Sigma}(z)) \tag{7}$$

whereby

$$IA_{S,c,\gamma,\Sigma}(z) = \sum_{j=1}^{k_{S,c,Det}(z)} i_{S,c,j} \cdot \Re\{\gamma(\tau_j)\} \cdot |\Re\{\gamma(\tau_j)\}|. \tag{8}$$

The term $k_{S,c,Det}(z)$ corresponds to the number of sample photons that hit the relevant detector section at scan position z and are coherent to the reference beam, $i_{S,c,j}$ represents their individual intensity and τ_j their individual optical path difference to the reference beam. The expression for $IA_{S,c,\gamma,\Sigma}(z)$ thus corresponds to the intensity-weighted sum of coherent sample photons combined with their respective contribution to the alternating interference signal component. The terms in (7) and (8) are expressed in this cumbersome way to avoid negative square root entries. A comparison of the idealized detector signal described in (2) and the photon-wise composition described in (4)-(8) is visualized in figure 6.

3.4 Coupling and Implementation

At this point, by adding white noise to the simulated signal, the signal formation of a virtual OCT measurement starting with the photon emission in the light source and finishing with the signal acquisition on the detector plane is fully described. Based on this description the simulations can be carried out by emitting a large number of photons and monitoring their trajectories and intensities until they hit the detector plane. With a time domain OCT (TD-OCT) method to be modeled in this paper, these calculations would have to be performed for each discrete cross-sectional height cut of the vertical scanning process. To reduce the computing effort of the virtual OCT measurement, the number of photon trajectories to be calculated must be reduced without reducing the validity of the simulation

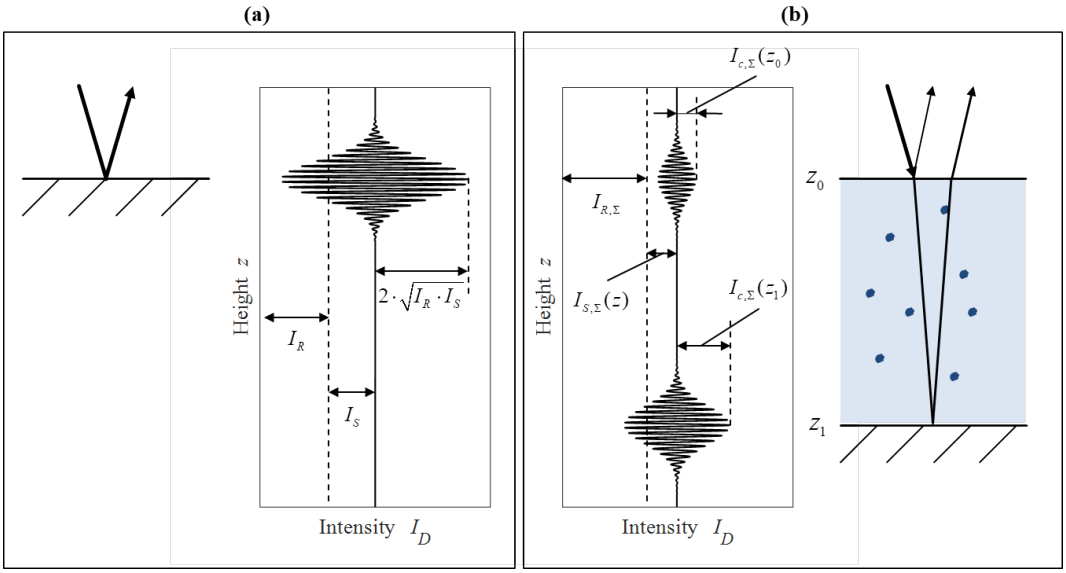


Figure 6 Illustration of the detector signal composition for (a) the idealized WLI measurement model in (2) (assuming specular reflection of light illuminating a mirror-like surface) and (b) the photon-wise composition model in OCT described in (4)-(8) (assuming partial reflection of light illuminating a layered tissue sample containing scatterers/absorbers).

result. This is done by exploiting the geometric conditions and by multiple adaptions of representative photon trajectories, which have been calculated once, thus drastically reducing the computing effort for the scanning process to be simulated by an upscaling method.

In the next section, the model is validated using real measurements performed by a reference instrument setup (which was initially designed as WLI) used as OCT. In the first step, the measurement conditions are provided to be idealized and thus should not be challenging from a measurement technology point of view. This is ensured by varying layer structures composed of only weakly light-scattering media (glass plates, air, specular reflecting surface). Based on this, the limitations of the current instrument setup will be explored by virtually increasing the difficulty of the measurement conditions. This will be done by virtually measuring thicker, optical rougher and more turbid (and therefore more challenging) virtual samples.

4 Validation of the OCT model

The first step towards validation of the OCT model is to create a virtual replica of the reference instrument. The reference instrument and the spectral properties of the (visual red) light source used for this purpose are visualized in figure 7. On the basis of this instrumental setup, an initial comparison of real and virtual measurements is carried out using three exemplary measuring tasks that are not very challenging in terms of measurement technology and comparatively easy to parameterize. The three measuring tasks under investigation are based on a simple bridge-like structure of identical glass plates on a specular surface as shown in figure 8. Thus, three differently structured measurement tasks can be taken from a single areal OCT measurement. These correspond to the following structures:

(a) specular reflecting surface (no layers to penetrate)
(b) glass layer – specular reflecting surface (one layer to penetrate)
(c) glass layer – air layer – specular reflecting surface (two layers to penetrate)

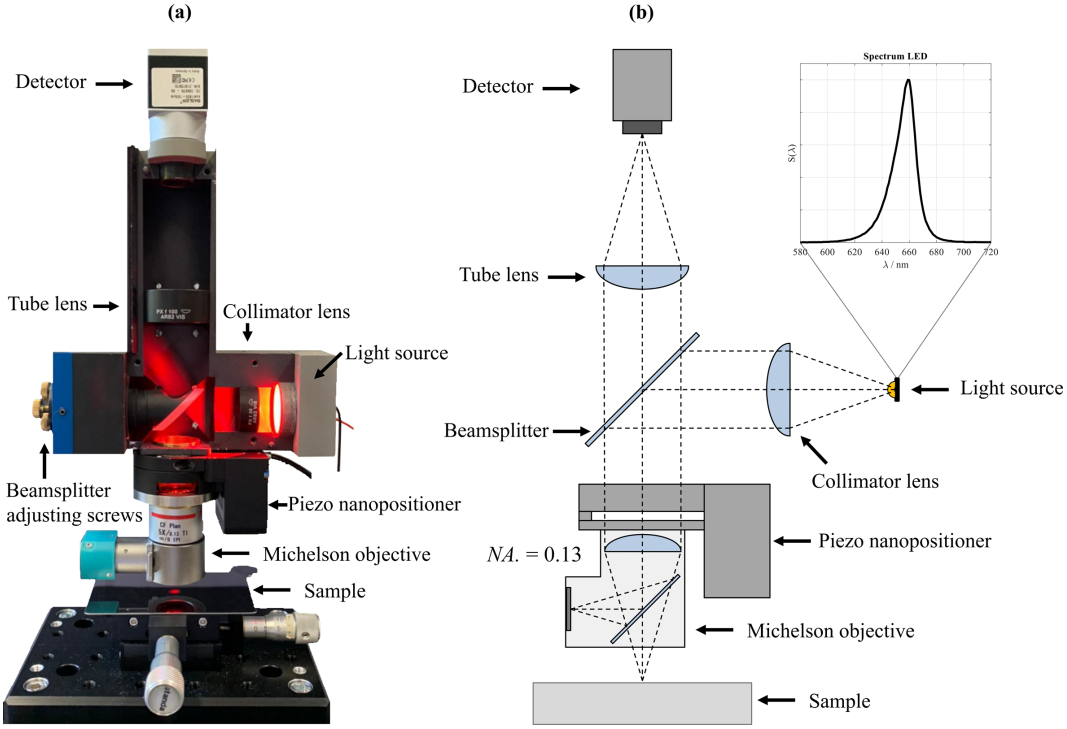


Figure 7 (a) Reference and (b) virtual instrument setup used for the validation of the OCT model.

Theoretically, a fourth measuring task (glass layer – glass layer – specular reflecting surface) could be extracted from the measurement setup visualized in figure 8, which is waived at this point since the optical height of this stack exceeds the scanning range of the piezo nanopositioner installed in the instrument.

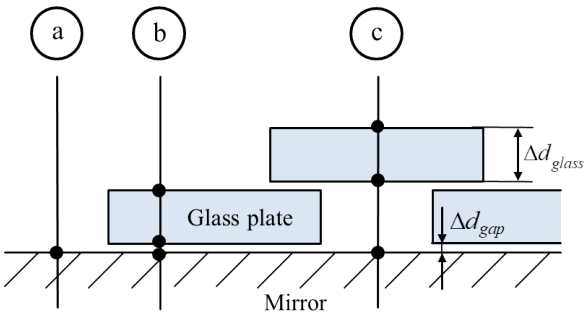


Figure 8 Measurement setup used to match virtual and real measurements. Labels 1 to 3 mark the three positions where the characteristic measurement signals were extracted from. The small black dots mark positions where the light interacts with boundary layers.

The simplicity of the parameterization of the optical material properties is based on the transparency of glass and air layers (no scattering or absorption, $\mu_s = \mu_a = 0$) as well as the (almost) ideal reflection properties of a flat mirror surface (ideally specular reflection, $g_{mirror} = 1$) provides the initial comparison of virtual and real measurements. The variable

Table 1 Estimated device and material properties of the measurement system provided by performed matching of virtual and reference measurement signals.

Device properties	
ρ	0.6
Material properties	
Δd_{glass}	106 µm
Δd_{gap}	31 µm
n_{glass}	1.53
$g_{surface}$ (optical surface roughness)	0.985

optical material properties are reduced to the refractive index (n_{glass}) and the optical surface roughness of the glass layer (here expressed by the anisotropy factor $g_{surface}$). Thus, the unknown (optical) instrument and material properties can be provided by performing a simple trial&error matching of simulated virtual measurement and the real measurement signals, which is done by comparison of respective signal amplitudes obtained. The matched signals are visualized in figure 9 and the unknown optical parameters determined on the basis of this procedure are summarized in table 1.

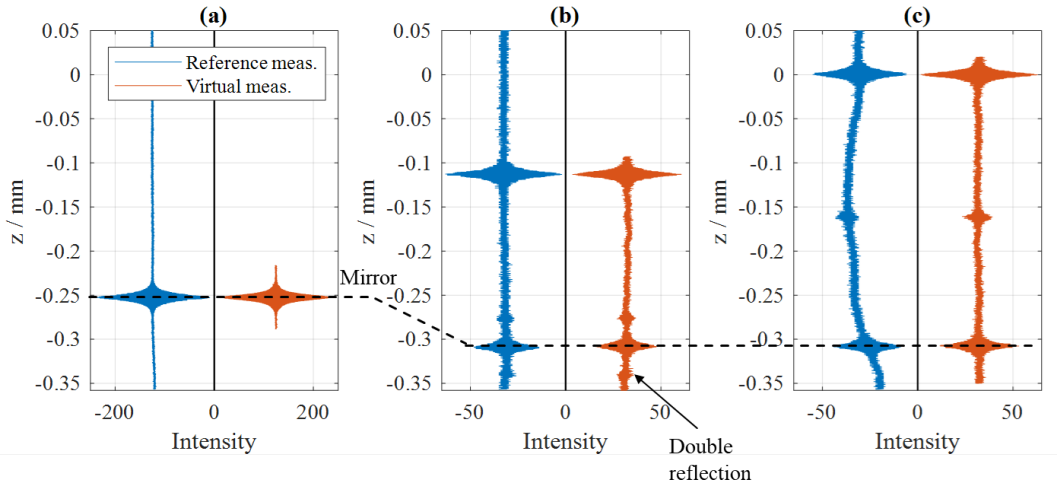


Figure 9 Comparison of measured (blue – left-hand side) and modeled (red – right-hand side) signals for the cases of measuring (a) the mirror surface, (b) the glass layer – mirror surface structure and (c) the bridge-like glass layer – air layer – mirror surface structure visualized in figure 8.

By setting the unknown modeling parameters in accordance to the values given in table 1, a high level of congruence between modeled and measured signals is achieved, which is imaged in figure 9. In the following, the physical limitations of the current instrument setup used as OCT are exemplarily explored using virtual measurements. For this purpose, a single-layered structure (tissue layer on a specular reflecting surface) is modeled as getting thicker ($\Delta z \uparrow$), optical rougher ($g_{surface} \downarrow$) and more turbid ($\mu_s \uparrow, \mu_a \uparrow, g \downarrow$) and thus the corresponding virtual measurement results are generated and evaluated.

Table 2 Instrument and material parameters chosen for the **full factorial** study of virtual measurements of the three cases (a)-(c) mentioned.

Instrument/ material parameter	(a) Variation of thickness	(b) Variation of optical roughness	(c) Variation of turbidity
NA	$[0.07 \quad 0.1 \quad 0.13]$	0.13	0.13
$\Delta z/mm$	$[0.05 \quad 0.1 \quad \ldots \quad 0.8]$	$[0.1 \quad 0.2 \quad 0.3]$	$[0.1 \quad 0.2 \quad 0.3]$
$g_{surface}$	0.985	$[1 \quad 0.99 \quad \ldots \quad 0.8]$	0.985
μ_s/mm^{-1}	0	0	5
μ_a/mm^{-1}	0	0	0.5
g	–	–	$[1 \quad 0.99 \quad \ldots \quad 0.8]$

5 Virtual measurements

The modeled setup for the virtual measurements, consisting of a single tissue layer placed on a specular reflecting mirror, is shown in figure 10. In the first measurement task to be

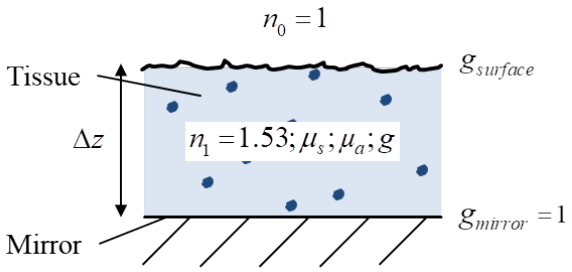


Figure 10 Measurement model of a single-layered tissue placed on a specular reflecting mirror.

modeled, the effect of a variation in the thickness Δz of the previously used glass plate on the measurement result is investigated given three Michaelson objectives with varying NA. Hereby it can be estimated where the limitations of the used instrument setup lie at almost ideal measurement conditions. Furthermore, it is investigated how a variation of the (optical) surface roughness, represented by the coefficient $g_{surface}$, affects the measurement result assuming other conditions to be constant and almost ideal. This describes, for example, the film thickness measurement of clear coatings which cause a certain surface roughness. This case is also investigated for variable film thicknesses. Finally, by varying scattering, absorption and anisotropy coefficients μ_s, μ_a and g, turbid tissue is modeled and the influence of these material parameters on the measurement result, assuming the layer thickness and surface roughness remaining constant, is investigated. This case corresponds, for example, to the layer thickness measurement of biofilms cultivated on smooth and thus specular reflecting surfaces. The instrument and material parameters chosen for the full factorial study of each of these cases are listed in table 2 and corresponding results of the virtual measurements are shown and evaluated in figure 11.

The investigation of the suitability of the current instrument setup for the various measuring tasks described above is based on the detectability of the signal component that can be assigned to the light reflection of the mirror. For this purpose, virtual measurements are performed and the generated intensity signals are evaluated by their envelope, which can be generated using Hilbert transform [31]. Since the geometrical and optical properties of

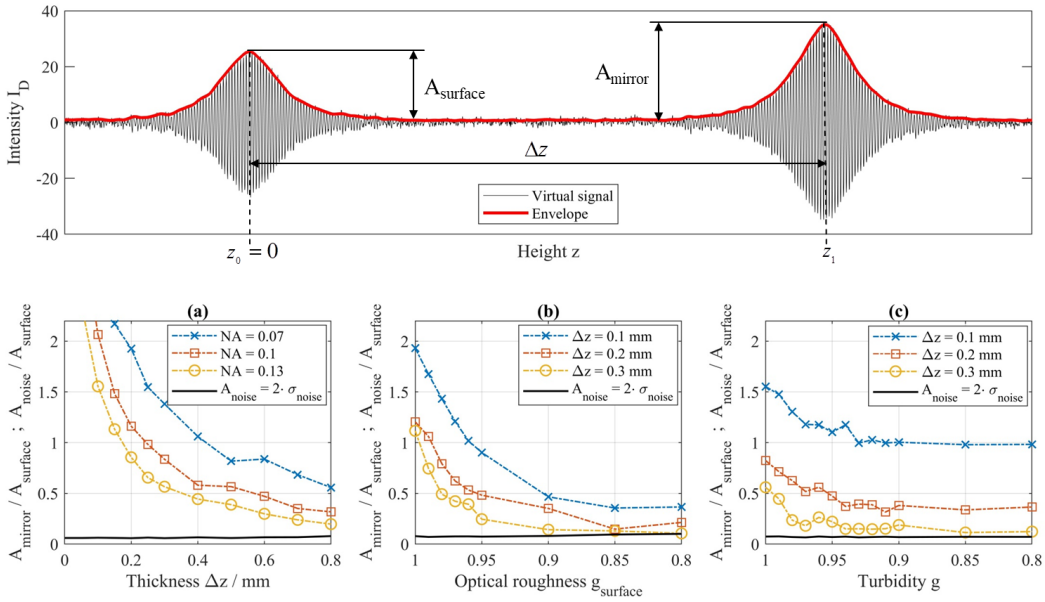


Figure 11 Results extracted from virtual measurements considering three different measuring tasks. Top: Exemplarily virtual measuring result and its envelope. Bottom: Evaluation of the virtually measured intensity signals using the parameters summarized in table 2 by plotting the amplitude ratio of mirror and surface reflection signal components against the disturbance variables (a) thickness, (b) optical surface roughness and (c) turbidity (represented by the anisotropy coefficient g). As a comparable measure of the instruments noise the ratio of $2 \cdot \sigma_{noise}$ and the amplitude of the surface reflection component is plotted as well.

the virtually measured sample match the dimensions of the experimental sample for which the OCT model was validated it is legitimate to evaluate the capabilities of the instrument to characterize samples of increasing level of complexity exclusively by performing virtual measurements. The upper part of figure 11 shows an example of a generated measurement signal including the generated envelope. By calculating the signal amplitude ratio between the intensity components of the surface and the mirror reflection while increasing the individual disturbance variables (layer thickness Δz, optical surface roughness $g_{surface}$, turbidity $\begin{bmatrix}\mu_s & \mu_a & g\end{bmatrix}$) and comparing it with the instrument's noise, which is also related to the surface reflection, the limits of the current instrument setup can be identified. The lower limit of the detectability is defined at this point by twice the standard deviation of the instrument's noise.

In general, it can be seen from the lower part of figure 11 that when the disturbance variables are increased, the intensity of the measured signal component reflected by the mirror decreases very sharply.

In the first case considered in (a), the dependence on the NA of the (Michelson) objective used is shown in addition to the strong dependence of the sample layer thickness. As described in [7], the dependence on the objectives NA can be explained by the displacement of the focal and coherence plane within the sample, which the reference beam does not experience and thus the ability for coherent interference is reduced. However, changing the lens actually used in this work ($NA = 0.13$) would not lead to an optimization of the instrument setup at this point, since the scanning range is limited to <400 µm. If this range is increased, however, the use of an objective with a smaller NA could be quite reasonable.

In the second case considered in (b) it can be observed that the optical surface roughness of the sample also has a strong influence on the signal quality. Due to increasing initial light scattering at the height level of the sample surface, the number of photons coherent to the reference beam hitting the coherence region (figure 5) decreases. According to the virtual measurements carried out here, even a small optical surface roughness ($g_{surface} < 0.9$) can cause the detectability of the signal component assigned to the mirror to fall below the noise level of the measuring instrument. A reduction of the noise level, e.g. by using a camera sensor system optimized for the light spectrum used, could remedy this situation.

In the third case considered in (c) it can be observed that especially the combination of turbidity and thickness of the tissue layer to be penetrated affects the quality of the measurement result. The decline in the signal amplitude ratio between surface and mirror reflection does not occur as abruptly with increasing turbidity as, for example, with surface roughness increments. More important in the context of measurability of turbid tissue is the "optical depth" (OD) [43], which is measured as

$$OD = (\mu_s + \mu_a) \cdot \Delta z. \tag{9}$$

6 Comments on a WLI&OCT convertible

Although WLI and OCT are based on the same measuring principle, the virtual measurements carried out above have shown that the current instrument setup can only be used as OCT for layer thickness measurements within highly restricted boundary conditions. This includes, for example, layer thickness measurements of thin, smooth and (partially) transparent materials, such as glass plates (see experiment), foils, transparent coatings or the detection of air inclusions, etc. The strong limitation of using the current instrument setup as OCT is due to several limiting factors. These include

- the height shift of the focal plane (lengthens within a refractive tissue) in relation to the coherence plane (shortens within a refractive tissue) of a focused light beam [7],
- the use of visual red light instead of near-infrared light,
- optical and technical components designed for the use of visual light (lenses, beam splitters, detector) and
- the maximum scanning range of <400 µm.

A very simple way to achieve an easy optimization of the current instrument setup towards application as OCT would be the use of an objective with low NA combined with an adjustable beam splitting ratio (both not available at the time of writing). By attenuating the intensity of the reference beam, the signal-noise ratio could be increased, which in turn increases the bandwidth of measurable structures. However, overall, the current instrument setup is not suitable for the use as OCT in context of diverse industrial applications, although the physical measuring principle corresponds to WLI.

7 Conclusion

Within this paper, a physical model of full-field time-domain optical coherence tomography (FF-TD OCT), which focuses the requirement of (virtually) measuring flexible layered structures, was developed, validated and exemplarily applied to investigate the extent to which the principles of classical WLI can be directly transferred to obtain layer thickness measurements by a using a simple low-cost instrument as WLI&OCT convertible. The developed model takes into account the optical and geometrical properties of the system

used (linear optics), as well as the light propagation in light scattering media (Monte-Carlo based modeling), and corresponding interference properties (partially coherent interference of light). The validation of the model was carried out on the basis of reference measurements. Three differently structured and highly controlled measurement tasks were used for this purpose. Subsequently, the knowledge gained from the validation was used to carry out virtual measurements of measuringt tasks with increased complexity (thicker layers, optically rougher surfaces, turbid tissue). On the basis of these virtual measurements, a first impression of the influence of various parameters and disturbance variables on the expected measurement result was given. It was shown that a currently existing low-cost WLI instrument setup cannot be used as OCT in general, but only to a very limited extend. The limitations are largely caused by the fact that the instrument was initially designed as WLI.

In future works, a FF-TD OCT optimized for industrial purposes will be realized with the aid of the OCT model developed in this paper.

Nomenclature and List of Abbreviations

Δd	layer thickness
Δz	height range, layer thickness
$\Delta\lambda$	full width at half maximum
γ	temporal coherence function
λ	wavelength of light
λ_0	average wavelength of light
μ_a	absorption coefficient
μ_s	scattering coefficient
ν	frequency of light
$\Re$	real part of complex number
ρ	beam splitting ratio
σ_{noise}	standard deviation of instrument noise
τ	optical path difference
A	amplitude
g	anisotropy factor
$g_{surface}$	optical surface roughness
i_0	initial intensity of an emitted photon
I_D	detector intensity
I_R	reference beam intensity
i_R	intensity of a single reference photon
I_S	sample beam intensity
i_a	absorption-related intensity loss
$I_{c,\Sigma}$	coherent intensity component
$I_{R,\Sigma}$	sum of reference photon intensities
$I_{S,\Sigma}$	sum of sample photon intensities
$i_{S,c}$	intensity of a single coherent sample photon
i_S	intensity of a single sample photon
$IA_{S,c,\gamma,\Sigma}$	intensity-weighted sum of coherent sample photons combined with their respective contribution to the alternating interference signal component
$k_{R,Det}$	number of reference photons hitting the detector
$k_{S,c,Det}$	number of coherent sample photons hitting the detector
$k_{S,Det}$	number of sample photons hitting the detector

l_c	coherence length of light
n	refractive index
NA	numerical aperture
OD	optical depth
r_R	reflectance of reference mirror
S	spectrum of light
z	height, scan position
CCD	charge coupled device
CSI	coherence scanning interferometry
FF	full field
FT	Fourier transform
LCD	liquid crystal display
MCML	monte carlo modeling of light transport
OCT	optical coherence tomography
SD	spectral domain
TD	time domain
WLI	white light interferometry

References

1 Max Born, Emil Wolf, and Avadh Behari Bhatia. *Principles of optics: Electromagnetic theory of propagation, interference and diffraction of light*. Cambridge Univ. Press, Cambridge, 7th ed., 11th reprinting edition, 2016.

2 Georgis Bulun, Matthias Eifler, Julian Hering, Georg von Freymann, and Jörg Seewig. Performance specification of areal surface texture instruments exemplified by a self-built wli. In Michael B. North Morris, Katherine Creath, and Rosario Porras-Aguilar, editors, *Interferometry XX*, page 15. SPIE, 2020. doi:10.1117/12.2567666.

3 William M. Cornette and Joseph G. Shanks. Physically reasonable analytic expression for the single-scattering phase function. *Applied optics*, 31(16):3152–3160, 1992. doi:10.1364/AO.31.003152.

4 Peter de Groot. Principles of interference microscopy for the measurement of surface topography. *Advances in Optics and Photonics*, 7(1):1, 2015. doi:10.1364/AOP.7.000001.

5 Peter de Groot and Xavier Colonna de Lega. Signal modeling for low-coherence height-scanning interference microscopy. *Applied optics*, 43(25):4821–4830, 2004. doi:10.1364/AO.43.004821.

6 Wolfgang Drexler. *Optical coherence tomography: Technology and applications /Wolfgang Drexler ... (eds.)*. Biological and Medical Physics, Biomedical Engineering. Springer, Berlin and Heidelberg, 2008.

7 Arnaud Dubois, editor. *Handbook of full-field optical coherence microscopy: Technology and applications*. Pan Stanford Publishing, Singapore, 2016.

8 Joy P. Dunkers, Richard S. Parnas, Carl G. Zimba, Richard C. Peterson, Kathleen M. Flynn, James G. Fujimoto, and Brett E. Bouma. Optical coherence tomography of glass reinforced polymer composites. *Composites Part A: Applied Science and Manufacturing*, 30(2):139–145, 1999. doi:10.1016/S1359-835X(98)00084-0.

9 Joseph W. Goodman. *Introduction to fourier optics*. W.H. Freeman Macmillan learning, New York, fourth edition edition, 2017.

10 Shengling Huang, Xin Wang, Yifan Chen, Jie Xu, Tian Tang, and Baozhong Mu. Modeling and quantitative analysis of x-ray transmission and backscatter imaging aimed at security inspection. *Optics express*, 27(2):337–349, 2019. doi:10.1364/OE.27.000337.

11 ISO 25178–70. Geometrical product specifications (gps) – surface texture: Areal —- part 70: Material measures (iso 25178–70:2014), 2014.

12 Steven L. Jacques. Modeling tissue optics using monte carlo modeling: A tutorial. In Steven L. Jacques, William P. Roach, and Robert J. Thomas, editors, *Optical Interactions with Tissue and Cells XIX*, SPIE Proceedings, page 68540T. SPIE, 2008. doi:10.1117/12.776997.

13 Paul Jansz, Steven Richardson, Graham Wild, and Steven Hinckley. Modeling of low coherence interferometry using broadband multi-gaussian light sources. *Photonic Sensors*, 2(3):247–258, 2012. doi:10.1007/s13320-012-0069-0.

14 Paul Jansz, Graham Wild, Steven Richardson, and Steven Hinckley. Simulation of optical delay lines for optical coherence tomography. In *2011 International Quantum Electronics Conference (IQEC) and Conference on Lasers and Electro-Optics (CLEO) Pacific Rim incorporating the Australasian Conference on Optics, Lasers and Spectroscopy and the Australian Conference on Optical Fibre Technology*, pages 1400–1402. IEEE, 2011. doi:10.1109/IQEC-CLEO.2011.6194128.

15 Boris Karamata, Markus Laubscher, Marcel Leutenegger, Stéphane Bourquin, Theo Lasser, and Patrick Lambelet. Multiple scattering in optical coherence tomography. i. investigation and modeling. *Journal of the Optical Society of America. A, Optics, image science, and vision*, 22(7):1369–1379, 2005. doi:10.1364/JOSAA.22.001369.

16 Boris Karamata, Marcel Leutenegger, Markus Laubscher, Stéphane Bourquin, Theo Lasser, and Patrick Lambelet. Multiple scattering in optical coherence tomography. ii. experimental and theoretical investigation of cross talk in wide-field optical coherence tomography. *Journal of the Optical Society of America. A, Optics, image science, and vision*, 22(7):1380–1388, 2005. doi:10.1364/JOSAA.22.001380.

17 Andrej Keksel, Matthias Eifler, and Jörg Seewig. Modeling of topography measuring instrument transfer functions by time series models. *Measurement Science and Technology*, 29(9):095012, 2018. doi:10.1088/1361-6501/aad467.

18 Andrej Keksel, Anna-Pia Lohfink, Matthias Eifler, Christoph Garth, and Jörg Seewig. Virtual topography measurement with transfer functions derived by fitted time series models. *Measurement Science and Technology*, 31(5):055008, 2020. doi:10.1088/1361-6501/ab5131.

19 Richard Leach. *Optical Measurement of Surface Topography*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. doi:10.1007/978-3-642-12012-1.

20 Siavash Malektaji, Ivan T. Lima, and Sherif S. Sherif. Monte carlo simulation of optical coherence tomography for turbid media with arbitrary spatial distributions. *Journal of biomedical optics*, 19(4):046001, 2014. doi:10.1117/1.JBO.19.4.046001.

21 Adrian Gh. Podoleanu. Optical coherence tomography. *Journal of microscopy*, 247(3):209–219, 2012. doi:10.1111/j.1365-2818.2012.03619.x.

22 Tuukka Prykäri, Jakub Czajkowski, Erkki Alarousu, and Risto Myllylä. Optical coherence tomography as an accurate inspection and quality evaluation technique in paper industry. *Optical Review*, 17(3):218–222, 2010. doi:10.1007/s10043-010-0039-y.

23 Fernando Puente León and Jürgen Beyerer. Oberflächencharakterisierung durch morphologische filterung (surface characterization by morphological filtering). *tm - Technisches Messen*, 72(12/2005):602, 2005. doi:10.1524/teme.2005.72.12.663.

24 Maik Rahlves and Jörg Seewig. *Optisches Messen technischer Oberflächen: Messprinzipien und Begriffe*. Beuth Pocket. Beuth Verlag GmbH, 1. edition, 2009.

25 Yogesh Rao, N. P. Sarwade, and Roshan Makkar. Modeling and simulation of optical coherence tomography on virtual oct. *Procedia Computer Science*, 45:644–650, 2015. doi:10.1016/j.procs.2015.03.121.

26 Stefan Roth and Achim Stahl. *Optik*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2019. doi:10.1007/978-3-662-59337-0.

27 Joanna Schmit. White-light interference 3d microscopes. In Kevin Harding, editor, *Handbook of Optical Dimensional Metrology*, pages 395–418. CRC Press, 2016.

28 Robert Schmitt, Friedel Koerfer, and Stephan Bichmann. Modellierung optischer messprozesse (modelling of optical measurement methods). *tm - Technisches Messen*, 75(4):6152, 2008. doi:10.1524/teme.2008.0832.

29 Robert Schmitt, Friedel Koerfer, Oliver Sawodny, Jan Zimmermann, Rolf Krüger-Sehm, Min Xu, Thorsten Dziomba, Ludger Koenders, Gert Goch, Andreas Tausendfreund, Stefan Patzelt, Sven Simon, Lars Rockstroh, Carsten Bellon, Andreas Staude, Peter Woias, Frank Goldschmidtböing, and Martin Rabold. Virtuelle messgeräte: Definition und stand der entwicklung (virtual measuring instruments: Definition and development status). *tm - Technisches Messen*, 75(5/2008):357, 2008. doi:10.1524/teme.2008.0872.

30 Heinrich Schwenke. *Abschätzung von Meßunsicherheiten durch Simulation an Beispielen aus der Fertigungsmeßtechnik: Zugl.: Chemnitz, Techn. Univ., Diss*, volume 36 of *PTB-Bericht F, Fertigungsmeßtechnik*. Bremerhaven Wirtschaftsverl. NW Verl. für neue Wissenschaft, Bremerhaven, 1999.

31 Jörg Seewig, Thomas Böttner, and Daniel Broschart. Uncertainty of height information in coherence scanning interferometry. In Peter H. Lehmann, Wolfgang Osten, and Kay Gastinger, editors, *Optical Measurement Systems for Industrial Inspection VII*, SPIE Proceedings, page 80820V. SPIE, 2011. doi:10.1117/12.889796.

32 Jörg Seewig, Matthias Eifler, Frank Schneider, Benjamin Kirsch, and Jan C. Aurich. A model-based approach for the calibration and traceability of the angle resolved scattering light sensor. *Surface Topography: Metrology and Properties*, 4(2):024010, 2016. doi:10.1088/2051-672X/4/2/024010.

33 Muhammad Faizan Shirazi, Kibeom Park, Ruchire Eranga Wijesinghe, Hyosang Jeong, Sangyeob Han, Pilun Kim, Mansik Jeon, and Jeehyun Kim. Fast industrial inspection of optical thin film using optical coherence tomography. *Sensors (Basel, Switzerland)*, 16(10), 2016. doi:10.3390/s16101598.

34 Lars Thrane. *Optical coherence tomography: Modeling and applications*, volume 1217 (EN) of *Risø-R*. Risø National Laboratory and Available from: Risø National Laboratory, Information Service Department, Roskilde, 2001.

35 Lars Thrane, Thomas M. Jørgensen, Mikkel Jørgensen, and Frederik C. Krebs. Application of optical coherence tomography (oct) as a 3-dimensional imaging technique for roll-to-roll coated polymer solar cells. *Solar Energy Materials and Solar Cells*, 97:181–185, 2012. doi:10.1016/j.solmat.2011.10.004.

36 Gerd R. Tillack, Christina Nockemann, and Carsten Bellon. X-ray modeling for industrial applications. *NDT & E International*, 33(7):481–488, 2000. doi:10.1016/S0963-8695(00)00019-0.

37 François M. Torner. *Entwicklung virtueller, optischer Sensoren zur Charakterisierung geometrischer Oberflächen*. Dissertation, Universität Kaiserslautern, 2018.

38 François M. Torner, Jayanti Das, Gerhard Stelzer, Barbara Linke, and Jörg Seewig. Fundamental analysis of the usability of an angle-resolved scattered light sensor for monitoring vibratory finishing processes based on ray tracing simulations. *Applied Mechanics and Materials*, 869:115–127, 2017. doi:10.4028/www.scientific.net/AMM.869.115.

39 François M. Torner, Gerhard Stelzer, Lukas Anslinger, and Jörg Seewig. Description and evaluation of a simplified model to simulate the optical behavior of an angle-resolved scattered light sensor. *Journal of Computing and Information Science in Engineering*, 17(2):31, 2017. doi:10.1115/1.4034386.

40 Andreas Tycho, Thomas M. Jørgensen, Harold T. Yura, and Peter E. Andersen. Derivation of a monte carlo method for modeling heterodyne detection in optical coherence tomography systems. *Applied optics*, 41(31):6676–6691, 2002. doi:10.1364/AO.41.006676.

41 Junxin Wang, Changgang Xu, Annica M. Nilsson, Daniel L. A. Fernandes, and Gunnar A. Niklasson. A novel phase function describing light scattering of layers containing colloidal nanospheres. *Nanoscale*, 11(15):7404–7413, 2019. doi:10.1039/C9NR01707K.

42 Lihong Wang, Steven L. Jacques, and Liqiong Zheng. Mcml—monte carlo modeling of light transport in multi-layered tissues. *Computer Methods and Programs in Biomedicine*, 47(2):131–146, 1995. doi:10.1016/0169-2607(95)01640-F.

43 Ashley J. Welch and Martin J.C. van Gemert. *Optical-Thermal Response of Laser-Irradiated Tissue*. Springer Netherlands, Dordrecht, 2011. doi:10.1007/978-90-481-8831-4.

44 Faming Xu, Haridas E. Pudavar, Prasad N. Prasad, and David L. Dickensheets. Confocal enhanced optical coherence tomography for nondestructive evaluation of paints and coatings. *Optics letters*, 24(24):1808–1810, 1999. doi:10.1364/OL.24.001808.

45 Min Xu, Thorsten Dziomba, and Ludger Koenders. Modelling and simulating scanning force microscopes for estimating measurement uncertainty: A virtual scanning force microscope. *Measurement Science and Technology*, 22(9):094004, 2011. doi:10.1088/0957-0233/22/9/094004.

Physics Simulation of Material Flows: Effects on the Performance of a Production System

Moritz Glatt[1]
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Bahram Ravani
Department of Mechanical and Aerospace Engineering, University of California at Davis, CA, USA

Jan C. Aurich
Institute for Manufacturing Technology and Production Systems, Technische Universität Kaiserslautern, Germany

Abstract

In cyber-physical production systems, material flows show complexity due to varying physical aspects of transported work pieces and autonomously selected transport routes. As a result, physically induced disturbances that may lead to delays or damages are hard to predict. The on-line usage of a physics engine offers potential to derive material flow parameters that enable safe transports with optimized accelerations. Previous work showed the feasibility of this approach and potential operational benefits through faster material flows. In consequence, the scope of this paper is to apply discrete-event simulation to investigate whether physics simulation of material flows leads to positive impacts on production system performance indicators such as throughput times and capacity utilization. The results indicate that increased velocity and acceleration of material flows can positively influence these indicators. In consequence, applying physics simulation to ensure safe transports with such high velocities and accelerations can improve the overall performance of a production system.

2012 ACM Subject Classification Computing methodologies → Modeling and simulation

Keywords and phrases Physics simulation, discrete-event simulation, cyber-physical production systems

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.15

Funding This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057.

1 Introduction

Modern production is largely influenced by cyber-physical systems (CPS), which contain embedded systems that interact with real processes through sensors and actuators. In addition, CPS interact both with the physical and the digital world and are connected with each other and in global networks [6]. The application of CPS within manufacturing leads to the term of Cyber-physical production systems (CPPS). In CPPS, CPS in the form of e.g. machines or material handling systems are linked within and across all levels of production, CPPS are a key feature of Industry 4.0 and enable flexible and adaptive manufacturing of customized products in small lot sizes [27]. The individual character of products is reflected in individual work plans and process sequences that are required for their production. In consequence, also the routes that individual products take through a CPPS as part of the material flow often vary. On these different routes, the physical behavior of the workpieces

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 15; pp. 15:1–15:26
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

during material handling might vary due to different aspects of the factory layout (e.g. ramps or curves). This complexity in the material flow, which is defined as the movement of discrete objects on transport ways or conveyors in steady and unsteady time intervals [3] is increased by product variations. As a result of customization, products can differ in terms of physical characteristics such as mass, inertia or surface roughness. Such differences increase the risk of physically induced disturbances that affect safety of employees or equipment as well as the operation of the production system. These disturbances (e.g. a workpiece that tips on a conveyor) are a result of the physical interaction between workpieces and material handling systems. In order to anticipate the described disturbances, material flows in industry are commonly performed at rather low velocities and accelerations. In this manner, the exposure of handled products and workpieces to dynamic influences (as described above) is minimized, since slow velocities and accelerations lead to small inertial forces during the material flows. In consequence, material flows take a relatively long time and potential for faster material flows often remains unexploited. Physics simulation is able to evaluate individual material flow constellations and to subsequently determine fast but safe material handling parameters such as velocity or acceleration. This approach bears potential to increase the velocity of different material flow processes individually, potentially resulting in reduced lead times and therefore an improved performance of the overall production system. In order to investigate this effect, the goal of this paper is to describe the fusion of a physics simulation model that can calculate optimal parameters on the material flow level with a discrete-event simulation model that simulates the operation of the entire production system. Based on this coupling, the effects of different velocity and acceleration configurations in a production system are quantified.

2 State of the art

2.1 CPPS

Advances in information and communication technologies have fostered the increasing implementation of CPS in several industries. Within a CPS, embedded systems, which monitor and control real physical processes by means of sensors and actuators, are connected to the global digital networks via communication facilities [6]. The embedded systems monitor and control real physical processes by means of sensors and actuators. If several CPS are networked within production, they form a CPPS. The networked elements of a CPPS consequently acquire information from their environment and act autonomously. A CPPS is thus able to react to internal and external changes [27]. Within the framework of the cooperative characteristics of a CPPS, elements such as machines, transport systems or operating resources exchange information independently. This results in adaptive, self-configuring and partially self-organizing production systems. CPPS can be seen as a measure of manufacturing companies to cost-effectively produce customized products that are increasingly demanded by customers. A key component of CPPS is the diversion from traditional, centralized control architectures. Traditionally, operations on the factory level were performed based on centralized planning. Since CPS allow connections as well as computation and on-demand control of actuators, even high-level decisions can be directed to individual entities on the shop floor level without further efforts. This enables decentralized control as well as high flexibility. Individual resources can communicate their current status and production tasks can be allocated to other resources instantly. Decentralized processing can be utilized in order to perform short-term scheduling [27]. One common control approach for of CPPS is agent-based control. In this method, each CPS-based manufacturing resource comprises a

software agent that communicates with the respective agents of other resources and work pieces in order to control production processes. Using these decentralized control approaches, manufacturing tasks can be assigned to production resources dynamically, which creates individual production sequences [40]. Often, redundant machines negotiate with products and among each other in order to self-organize the manufacturing sequences of varying products [41]. In consequence, each product can take different routes through the CPPS. This is depicted in Figure 1 by showing the routes of example material flows through a CPPS. Material flow complexity can result from the number of workstations that are passed for one product, as well as from the variance of paths: The routes that workpieces and products take are not known before the actual execution due to short-term routing and redundant workstations. On these varying routes, different physical influences such as curves or ramps might occur. As a result, even the identical product variants might show different material flow routes and different physical influences. In this context, CPPS show similarities and

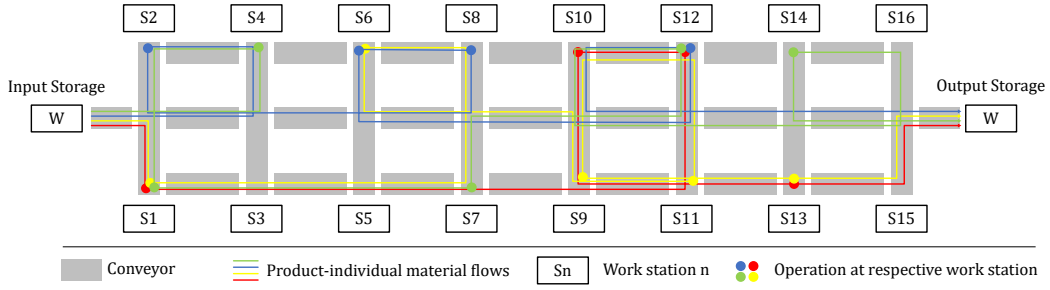


Figure 1 Varying routes in CPPS [19].

utilize common principles as flexible manufacturing systems (FMS) or matrix manufacturing systems (MMS). FMS comprise universal workstations (e.g. machine tools or assembly machines) and material handling systems to enable the flexible production of mid-volume and mid-variety production. FMS are capable of processing a variety of different part variants belonging to the same family simultaneously at the various workstations, and the mix of part styles and quantities of production can be adjusted in response to changing demand patterns [11]. Similarly, MMS are composed of modular workstations that are connected by a flexible transportation system. Through redundant workstations, the individual routing of the material is not known beforehand [13]. Both concepts provide flexibility regarding the manufactured products based on their architecture and composition. CPPS add to this flexibility by enabling decentralized control through the enabled communication between machines. Therefore, CPPS are often structured according to the principles of MMS or FMS, which is also the case in this study. Summarizing, CPPS are suitable for the economic production of small lot sizes that are a result of customization. Amongst other aspects, this flexibility is achieved through agent-based, decentralized control. These two characteristics impose challenges on the physical material flows, which are elaborated in the next subsection.

2.2 Material flows in CPPS

As described, meeting customer demands for individual products is one of the key requirements that today's production systems and thus CPPS need to fulfil. Product variety leads to a wide span width of product variants, in basic models as well as in variants within the models [43]. Depending on the extent of customization, objects (both finished products or workpieces)

that are handled within material flows may have varying physical attributes like shape, size, mass, inertia or surface roughness [8]. In consequence, the interaction between these objects and material handling systems may vary as well. Furthermore, even identical products may pass through CPPS on different routes where they can experience varying physical influences due to characteristics of the factory layout (e.g. inclinations or curves) [16]. As a result of these influences, the physical behavior of each material flow process in CPPS may become highly individual. The physical behavior of processes in material handling systems that can lead to the described disturbances is determined by certain operating parameters (e.g. the acceleration or the torque of the actuators). These parameters however often not only influence the physical behavior, but also affect the performance of the material handling system and therefore the overall production system. For instance, if a material handling system is operated with higher velocities and accelerations, this will lead to the positive effect in terms of faster material flows and shorter throughput times [22]. However, as a tradeoff, this measure also imposes higher inertial forces on the transported workpieces, which increases the chance of disturbances as described earlier. The goal of reduced disturbances while achieving high transport accelerations and velocities could be accomplished by fixtures for load securing. However, in case of customized products, these mechanisms need to provide a high degree of adaptability which leads to high cost and complexity. Furthermore, load fixing increases the material handling time and therefore throughput time, especially if a high number of transport processes is required. Another frequently used solution is to perform material flows with slow velocities and accelerations to exclude the possibility of disturbances. As a result, most transports in today's industrial material handling systems are performed slower than necessary, which leads to the longer transport durations. A promising approach to overcome these disadvantages, is to automatically select operating parameters of material handling systems according to type and characteristics of the transported load [2]. This requires special simulation techniques that would involve physical modeling. Discrete-event simulations that are commonly used to simulate material flows on the production system level do not allow to find physically suitable parameters. Instead, a possible approach to maximize the performance of the material handling system while preventing disturbances is the usage of physics simulation to simulate the physical interaction between workpieces and material handling systems during every individual material handling process. Performing physical simulation requires a deeper understanding of the mechanics that is of relevance during material flows. As described in [18], material flows are mostly performed with horizontal movements. This is the case in many common material handling systems like automated guided vehicles (AGVs) or conveyors. The physical behavior of an object that is carried on such a material handling system can be described as a mechanical system as depicted in Figure 2. Frequently, carried objects are held in place by a frictional contact between the objects surface and the material handling systems load bed. This common application scenario is shown in the figure. This requires a deeper understanding of the mechanics that is of relevance during material flows. As described in [16], material flows are mostly performed with horizontal movements. This is the case in many common material handling systems like automated guided vehicles (AGVs) or conveyors. Frequently, carried objects are held in place by a frictional contact between the objects surface and the material handling systems load bed. The physical behavior of a carried object in this common application scenario is shown in the simplified mechanical system in Figure 2: The object, indicated by a grey box has a mass that leads to a resulting gravitational force F_g. External forces F_{ext} and Moments M_{ext} of varying cause can apply. All dynamic influences lead to a resulting Force F_{res} and a resulting Moment M_{res} that act on the center of mass of the regarded object and define the physical behavior of this object.

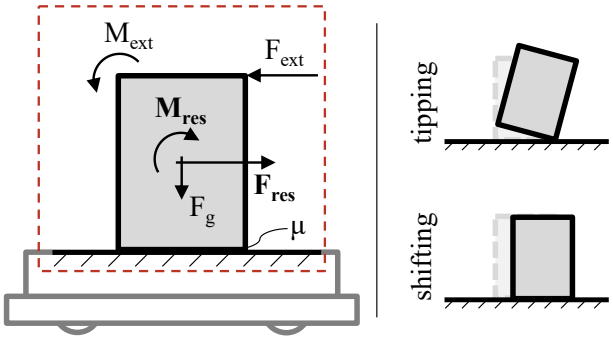


Figure 2 Mechanical system of material handling (after [18]).

Regarding horizontal material flows, disturbances occur, when the transported work piece is dislocated from the material handling system, which induces e.g. shifting or tipping. This applies, when the magnitude of F_{res} and M_{res} outweigh frictional and gravitational forces that hold the object in place. The occurrence and the extent of such disturbances are a result of the constellation of the following influencing factors:

- **Mass distribution of the workpiece:** The inertia of the work piece as well as the location of its center of mass. This is often a result of the aspect ratio of the work piece. In general, tall work pieces with a large height-to-width (in movement direction) ratio tend to tip, while flat workpieces are more likely to shift.
- **Friction between the workpiece and the material handling system:** The higher the friction coefficient μ between the workpiece surface and the material handling system at the respective contact surface, the less likely is the workpiece to shift.
- **Accelerations:** According to Newton's second law of motion, accelerations of the material handling system cause a force which applies at the center of gravity of the workpiece. This may result in tipping or shifting.
- **External forces and moments:** Furthermore, the application of external forces and moments may cause disturbances as well (e.g. collision with factory infrastructure).

Based on these fundamental mechanical correlations, several disturbances in industrial practice may occur: Workpieces can fall off a conveyor or tip over because of acceleration forces [44]. Workpieces that have fallen or otherwise changed their orientation on a conveyor can induce jamming of the material handling system and result in downtimes [20]. Therefore, material handling systems in industry need to be stopped smoothly to prevent the tipping of transported objects. This removes the necessity for additional load securing [36]. Material handling systems when considering the above problems are often operated at rather low velocities. For instance, automated guided vehicles (AGV) usually move with velocities in the range of 1 m/s, which is slower than a walking adult. Also, stationary conveyors like transfer systems operate at comparably low velocities in the range of 0.2-0.3 m/s [10, 5]. An advantage of these slow velocities is that they minimize the dynamic influences on the transported workpieces. Slow transport mostly involves slow accelerations, which in turn reduces the inertial forces that act on the workpieces during accelerating and stopping. Also, centrifugal forces in curves are mostly noncritical in these velocity ranges. In consequence, the risk of physical material handling disturbances is comparably low. However, material handling is estimated to cause about 40-80% of all operating cost within a production system [37]. It is considered as a non-value adding element during the manufacturing of a product. Non-stationary material handling systems are subject to the described disturbances as well.

Automated guided vehicles (AGVs) enable flexible routing and are therefore frequently used in manufacturing. Maintaining load stability is crucial for AGVs, because loads can tip in narrow curves due to centrifugal forces or during emergency stop situations as a result of the load's inertia [34]. Especially in environments that are shared with employees, AGVs must have the ability to stop within a safe distance, which usually defines the maximum AGV velocity [23]. Therefore, AGV velocity is mostly limited to values around 1 m/s to generally minimize the risk of loads falling off due to inertia or centrifugal forces [24]. Besides AGVs, loads falling off fork lifts are responsible for a majority of fatal injuries in industrial environments, often resulting from operators misjudging the varying physical attributes of the transported load in combination with the dynamic behavior of the vehicle [38]. As these examples show, physical attributes of loads can cause disturbances resulting from the interaction with different material handling systems, often resulting from horizontal accelerations. The result can be accidents, damage to products or to factory infrastructure, increased downtime of material handling systems, longer delivery times and additional cost. Considering the fact that material handling accounts for about 50% of all industrial injuries [37] and constitutes 15% to 70% of the manufacturing cost of a product [12], preventing the described disturbances bears potential to increase safety and economic efficiency of a production system. Summarizing, disturbances that result from the physical interaction of material handling system operation and the transported workpiece can lead to damages and delays up to injuries of staff and are therefore to be prevented. Previously, most material flow systems used to handle a limited number of products and workpieces. In consequence, if a certain configuration led to disturbance-free operation of the material handling systems, this configuration could be applied for all upcoming material flows with the same workpieces and routes. This is often a result of trial and error. However, as stated above, both product and flow variance can make every material flow unique and unprecedented in CPPS. Thus, this approach fails at determining suitable parameters, since new products require new evaluations. Therefore, in order to perform material flows as fast as possible, the physical constellation of every individual material flow has to be evaluated. For this purpose, physics simulation based on a physics engine offers potentials.

2.3 Physics simulation

The described effects that can lead to disturbances are based on dynamics and kinematics of rigid bodies (e.g. the workpieces and the respective elements of the material handling system). These phenomena can be simulated with physics engines [26], which is from now on referred to as physics simulation (also often called physics-based simulation). Physics engines are computer programs that allow a user to simulate and quantify the behavior of a set of rigid bodies over time as a result of external forces. Having their roots in computer graphics with the initial aim to efficiently compute physical occurrences for computer generated animations (e.g. video games), physics simulations have increasingly been applied in engineering applications. The advantage of using them is that after providing a set of defined constraints and physical attributes, many situations can be simulated without additional modeling efforts [18]. For example, after initial modeling of a material handling system, upcoming material flow processes can be simulated automatically by incorporating respective data about the transported objects and the adjustment of respective parameters. The basic workflow of physics simulation is as follows: After initialization, the considered objects (rigid bodies) are loaded into the simulation scene. Typically, these objects are represented by separate polygon meshes defined by the vertices, edges and faces of the individual object. These meshes can be derived from computer-aided design (CAD) drawings

or specifications of workpieces. Before simulating, each object is assigned with physical properties like mass, inertia or friction coefficients. After definition of the properties, the interaction between the objects is simulated in discrete time steps [9]. Figure 3a) shows a typical simulation loop that is performed during each of these time steps of a physics simulation. In this Figure, first, the simulation detects any collisions between the objects (collision detection). For this purpose, the described mesh of every individual object serves as its respective collision geometry. After collisions have been detected, the respective contact points are registered. With respect to these contact points, physics equations regarding the motion of the individual objects are defined and solved in order to determine contact forces. This step helps to achieve proper friction simulation and to prevent objects penetrating each other and is referred to as contact handling. In the subsequent step, collision resolving handles different kinds of collisions: Contacts that have not occurred in previous time steps imply collisions that are accompanied by impulsive forces. These forces cause instant change of object velocities and are often handled separately from pre-existing contacts (e.g. an object resting on another). Finally, when all contact forces have been computed, the new positions and velocities are calculated via time integration before the described loop is performed again [4]. This is typically done with a frequency of at least 60 Hz. Due to the functional principle of physics simulation, the physical behavior of material flow processes can be simulated. In consequence, physics simulation can simulate and predict a range of different disturbances, as long as they are a result of those physical interactions. Examples for this type of disturbances are described in Section 2.2. Physics simulation has been applied in the past for simulation of material flows for virtual commissioning of planned production systems prior to their operation (see, for example, Zäh et al. [44], Reinhart et al. [33], Hoher et al. [21] or Alkan et al. [1]). It was indicated that using physics simulation to simulate material flows during the operation of a production system could result in decreased lead times [15] and reduced downtimes of material handling systems [44]. However, none of the existing approaches have proved a quantitative evaluation of such conclusions. In our previous work, the authors addressed this research gap with the aim of utilizing physics simulation to support the operation of material handling systems. The python library pyBullet was selected as a suitable environment to implement the envisioned simulation model [7]. The library pyBullet provides an extensive and adaptable framework to simulate the mechanical interaction of rigid bodies. Results of our previous work indicated that pyBullet is able to reliably predict the required types of phenomena. The resulting physics simulation model of the authors allowed to accurately simulate physical phenomena during material handling. This model was integrated into a digital twin concept which enables determination of suitable acceleration values for certain transported workpieces. This is done with a simulation model of an exemplary conveyor, shown in Figure 3b). The approach showed the potential of decreasing the required transport times with respect to the workpiece's physical attributes [19]. Summarizing, the simulation of material flows prior to their execution offers the potential of performing material flow processes faster and without load securing. By estimating respective parameters (e.g. velocity or acceleration), loads can be transported with increased yet safe velocity. This may result in improved material handling system utilization and decreased lead times. However, the systematic effect that this approach has on a production system level has not yet been investigated. For this purpose, the use of discrete-event simulation in combination with physics-based simulation is investigated in this paper.

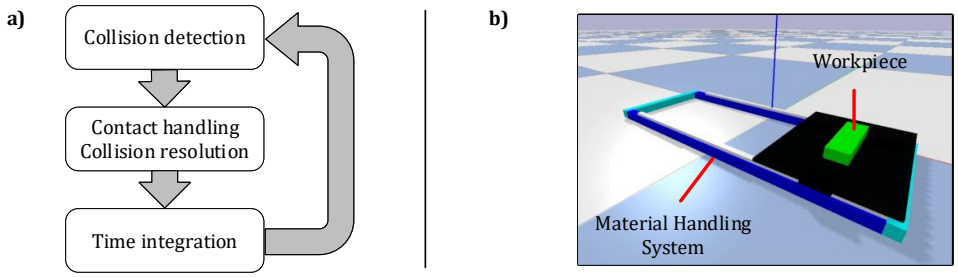


Figure 3 a) Simulation loop of a time step in physics simulations [4]; b) Physics simulation showing a workpiece on a conveyor [19].

2.4 Discrete-event simulation

Discrete-event simulation is a simulation technique, in which the temporal evolution of a system is modeled via state variables that change instantly at certain times. At each instant of time, the occurrence of certain discrete events changes the state of the overall system [30]. This allows modeling and simulation of the operation of complex systems and the tracking of individual resources. Various performance indicators can then be used to provide a comparison between different system parameters or alternatives, for example with regards to a system configuration. Discrete-event simulation is, therefore, suitable for simulating complex production systems [31] both in the planning as well as in the operational stage [29]. Several review papers (e.g. [29] and [28]) list numerous approaches that apply discrete-event simulation within manufacturing, e.g. for evaluating control strategies or to plan factory layouts. No approach however evaluates the performance of a production system with regards to the variation of physical attributes like velocity or acceleration within the material flow. Furthermore, besides a conceptual outline of the authors [17], no approach combines discrete-event simulation with physics simulation in order to analyze the effects of the latter on the performance of a production system. In current literature, few studies investigate the effect of material flow acceleration or velocity on the production system performance: Um et al. [39] incorporate AGV acceleration and velocity in the simulation of an FMS and conclude that especially velocity influences the throughput. Filz et al. [14] also vary AGV velocity within a matrix manufacturing system in their simulation study. The results indicate that the utilization of the manufacturing system increases with bigger AGV velocities. In these studies, the physics of material flows is however neglected and the physical applicability of the chosen velocities and accelerations is not considered. In addition, evaluating the effects of varying accelerations is not investigated as all previous studies regard the same values for all transports within one simulation scenario. Besides a conceptual outline of the authors [17], no previous approach combines discrete-event simulation with physics simulation in order to analyze the effects of physics-simulation-based individual accelerations on the performance of a production system.

2.5 Research Gap

After reviewing the state of the art, it can be concluded that characteristics of CPPS may lead to physically induced disturbances within material flow processes. This can result in accidents and can hamper the operation of a CPPS. Using physics simulation, the authors have shown

that individual material flows can be simulated with the purpose of determining fast but safe acceleration and velocity values [19]. A promising approach is, therefore, to simulate each material flow process prior to its execution with the purpose of finding the maximum safe acceleration. The effect of these measures on the performance of a production system need to be investigated for the following reasons: Higher velocities and accelerations obviously decrease the time that is needed to transport one object from one place in the production system to another. The overall performance of a manufacturing system is however affected by more factors than just transport time (e.g. wait times). Changed velocities and accelerations might affect the behavior of the entire material flow, which can even influence other aspects such as wait time in buffers. Furthermore, product-individual material flow speeds and velocities omit the commonly chosen practice of selecting one equal speed configuration for all material flows. It needs to be analyzed how this inhomogeneous distribution of velocities and accelerations affects the overall production system performance. For the purpose of analyzing effect of certain organizational configurations on a production system, discrete-event simulation has been widely applied in manufacturing. This paper aims at addressing this research gap by performing a simulation study that combines physics simulation with discrete-event simulation. The approach to address this question is elaborated in the next Section.

3 Approach

This Section describes the approach that is chosen to investigate, whether the application of physics simulation can lead to improved production system performance, which is expressed through the measurement of throughput times and conveyor utilizations. Therefore, the approach consists of two simulation models (see Figure 4):

- A **discrete-event simulation model** is implemented that contains the structure and the control architecture of a customized small series production system. On this foundation, the operation of a production system over a certain amount of time can be simulated and different scenarios can be compared. The objects of interest in this case are not the value-adding process such as manufacturing or assembly, but the material flow processes in between these steps. The kinematical characteristics of these material flows, e.g. velocity and acceleration are to be varied in different scenarios. Finally, the effects that these different scenarios have on throughput time and conveyor utilization will be quantified.
- The discrete-event simulation is enhanced with a **physics simulation model** that is able to simulate the physics of the material flows in the production system. For this purpose, the maximum safe accelerations of all workpieces that are transported in the production system are determined. The resulting parameters are then transferred to the discrete-event simulation model.

The following Section describes the application of this approach in a simulation study.

4 Simulation study

4.1 Characteristics of the considered production system

The considered production system produces parts of transmission systems, which includes shafts, gears, plates, housings and blocks. Due to customization, a total range of 50 parts within these categories is being manufactured. All parts vary in terms of their physical attributes as a result of different shapes, mass distributions and surface structures. Furthermore, the parts require different process sequences. To enable this flexibility, the

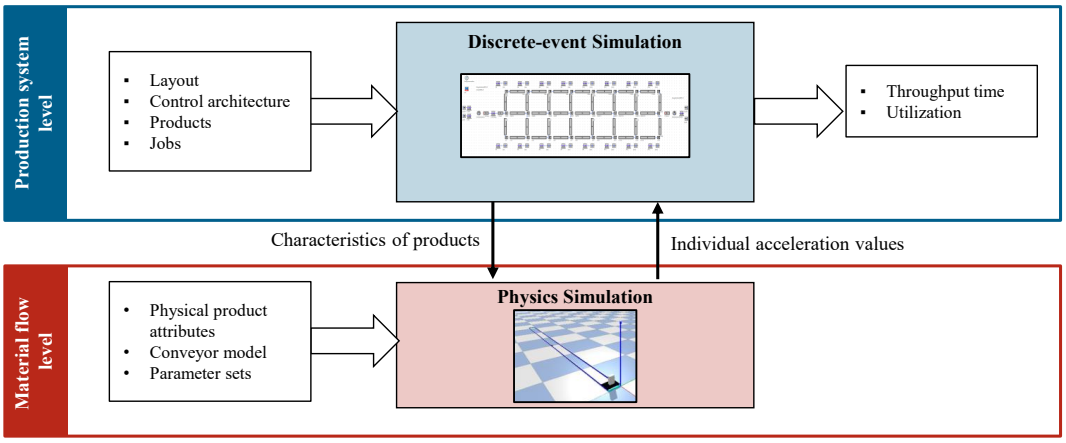


Figure 4 Approach.

production system consists of several workstations with universal processes like milling, turning or assembly. In the production system, an agent-based control architecture is implemented. Using an agent based control, parts, workstations and conveyors automatically find suitable production sequences. This makes the considered production system a CPPS [40]. Further flexibility is introduced through the material flow: The workstations are connected via a network of conveyors. These conveyors transport individual workpieces on workpiece carriers. The conveyors are driven by electric motors and all conveyors have a length of 6 m. When a workpiece is being transported, the respective conveyor accelerates the workpiece carrier with a constant acceleration "a" until a desired constant conveying velocity "v_c" is reached. At the end of the conveyor, the pallet is decelerated with the same deceleration "-a". The resulting kinematic characteristic of the conveyors is shown in Figure 5. The graph shows that with increased accelerations, v_c is reached faster, which leads to a higher average transport velocity and shorter transport times. This difference in required transport time between a = 1m/s^2 and a = 3m/s^2 (v_c = 2m/s in both cases) is indicated in the figure.

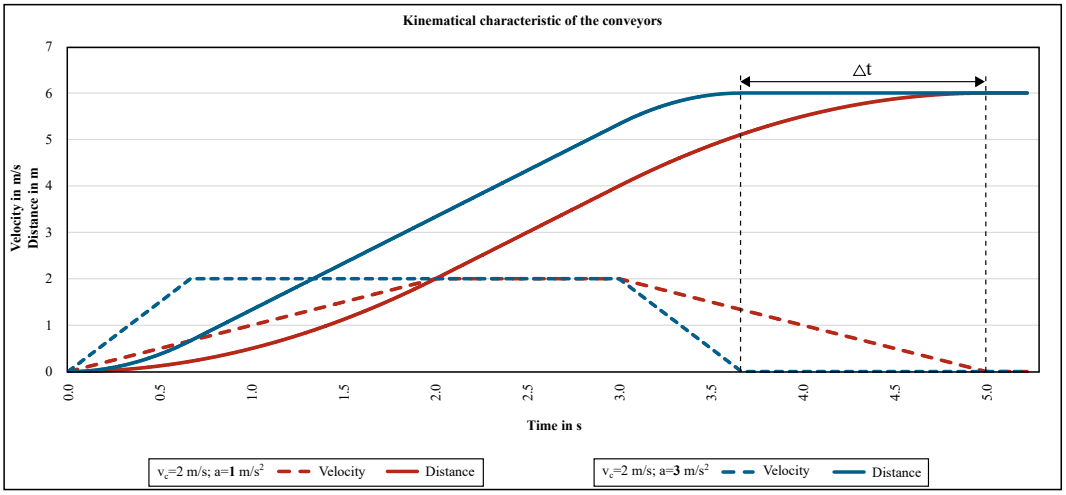


Figure 5 Kinematical characteristic of the conveyors.

4.2 Discrete-event simulation model

Implementing the discrete-event simulation included both the structure as well as the control architecture of the production system.

4.2.1 Structure of the discrete-event simulation model

The production system in the simulation model is a CPPS, in which the production of individual transmission components takes place. It is assumed that five fundamentally different product types are produced in the production system under consideration: Block, gear wheel, housing, plate and gear shaft. These product types are mutually characterized by different functions and geometries. Within a product type, the components are designed according to the customer's individual requirements, so that a distinction can be made between different characteristics. Thus, each component of a particular product type in turn has individual properties such as height and width. The product type distribution is listed in Table 1.

Table 1 Used part types and distribution.

Part type	Number of individual components within part type
Block	10
Gear wheel	11
Housing	10
Plate	6
Gear shaft	13
Sum	50

The individual components influence the selection of a motion profile, since the calculation of a component-specific acceleration is possible based on its geometry and other physical attributes like e.g. the friction coefficient between the part and the conveyor. During the simulation, each of the 50 individual components is produced four times in the production system under consideration. Here, production is carried out with batch size 1, so each job enters the production system at a different point in time. Consequently, a total of 200 jobs that lead to 200 individual components are performed in the simulation study. Furthermore, the components also influence the sequence of the steps to be performed. The production system comprises eight different production processes: Milling, turning, drilling, grinding, deburring, washing, forging and assembly. Each job is randomly assigned to one of 16 workstations (WS). Table 2 gives an overview over the WSs in the production system and the respective capabilities of the respective WS. For example, "Milling" can be performed on WS 1, WS 3 and WS 8, while "Washing" is exclusively performed on WS 9.

In order to enable transport between the individual WSs, networked conveyors are used. Customer-specific production with different routings and correspondingly different routes through the production system requires the use of decentralized control. Accordingly, the conveyors themselves determine the route that the respective workpieces take through the system. The workpieces are placed on a flat workpiece carrier for transport, which in turn moves through the production system on conveyors. Since individualized fixtures or recesses for inserting and securing the workpieces are very costly in customized production and the alternative of fixing the workpieces on the workpiece carrier takes a certain amount of time, the components are instead secured by selecting the suitable acceleration profile. Consequently, the appropriate values for acceleration have to be chosen with regards to

Table 2 Processes and respective workstations.

Process	workstation
Milling	WS 1, WS 3, WS 8
Turning	WS 2, WS 4, WS 5
Drilling	WS 6, WS 7,
Grinding	WS 12
Deburring	WS 11
Washing	WS 9
Forging	WS 10
Assembly	WS 13, WS 14, WS 15, WS 16

the transported workpiece to prevent it from tilting or slipping off the workpiece carrier without securing them. All conveyors that connect the WSs are assigned the length of 6 m. The conveyors are connected by transfer devices, which can move the loaded workpiece carriers to the next conveyor as well as to the adjacent WS. Since only one workpiece carrier can be on a conveyor at a time, buffers are placed in front of the respective WSs to avoid conveyor downtimes due to blocking. The buffers follow the first-in-first-out principle and transfer the workpiece with the longest waiting time immediately when the corresponding WS becomes vacant. A simplified overview of the production system is depicted in Figure 6. As shown in this Figure, individual WSs are arranged in an O-shape connected by a system of rectangularly structured conveyors. Individual stations with odd numbers are located in the lower area, whereas the remaining stations are located in the upper area. A centrally running conveyor line and cross-connections between the individual stations enable a greater variety of transport routes. The workpieces enter the production system through the source (left) and leave it again via the sink (right) after completing the last production step.

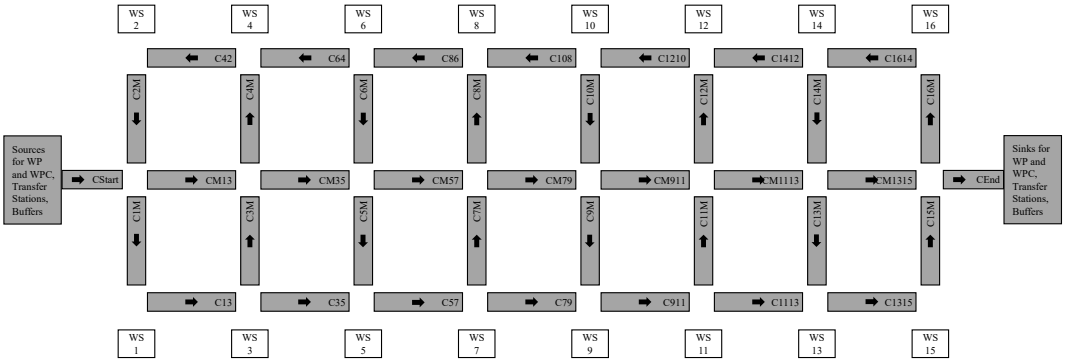


Figure 6 Simplified overview over the production system structure (transfer stations and buffers not included).

The described production system is implemented as a runnable discrete-event simulation model with the help of the simulation software SIEMENS Plant Simulation version 15.0 [35]. The implemented structure of the production system in the simulation model is shown in Figure 7. In order to design a runnable model from the simulation model, it is further necessary to control the working steps to be performed. This is outlined in the following subsection.

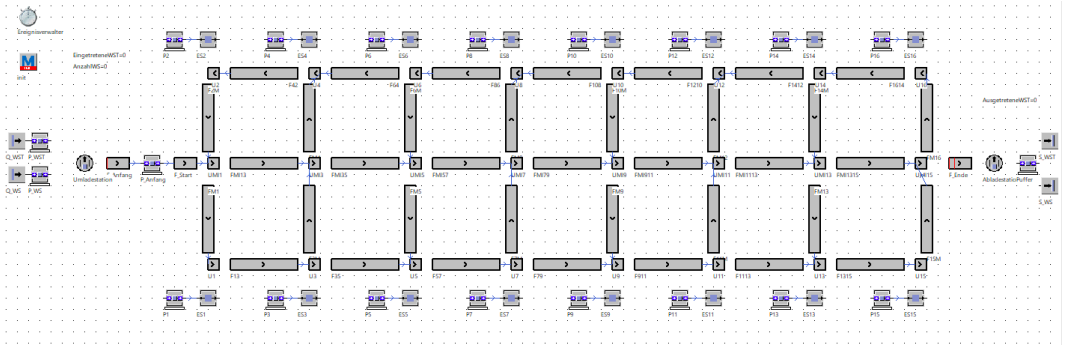


Figure 7 Structure of the production system in the implemented discrete-event simulation model.

4.2.2 Control of the discrete-event simulation model

To control the processes of the described production system, several methods and tables were added to the described simulation environment. In this context, methods refer to pieces of executable code that are being triggered by certain events (e.g. a conveyor passing a certain point), while tables are used to provide numerical values and other information that is required to run a discrete-event simulation. Plant Simulation requires different methods and tables which ensure the processing of the component-specific work plan as well as the coordination of the material flows. For the sake of clarity, all tables are preceded by the letter "T" and all methods by the letter "M". In the following, all methods that are utilized in the simulation model are explained. In addition, the most important method for the control is described with the help of a flow chart. The methods can be integrated as input or output control modules of the individual blocks and are executed before the workpieces enter or leave the corresponding block. In this context, a module is understood to be a source, sink, an individual station, a conveyor, a converter as well as a transfer or an unloading station. The components first enter the production system through the source S_{WS}. For this purpose, each of the 200 components is randomly assigned an entry time in the table T_{Source} using the method M_{init}, for which an equally distributed random number is generated in the interval from 1 min to 200 min. This means that on average one new production order with a workpiece to be produced enters the system every minute. The workpiece carriers that are necessary for the transport of the workpieces are generated within the first 100 s in the source S_{WPC} and then wait in buffer B_{WPC} for a workpiece to be loaded. As soon as a workpiece is generated, the method M_{WPC} provides a workpiece carrier on which the workpiece is transported through the production system. The workpiece is loaded onto the workpiece carrier via the reloading station and the conveyor C_{Start}. In combination with method M_{WP}, the output control M_{CStart} ensures that the name of the loaded workpiece can be accessed. To do this, the name is entered to the T_{count} table, which is necessary for processing the individual work plan. The workpiece and its carrier wait in the buffer B_{Start} until the first conveyor section C_{Start} is free and both can enter the actual production system. When they enter, the number of workpieces in the system n_{WP} is increased by one using the method "M_{nWP}" in order to improve the traceability of the production process. In the following, the terms "workpiece" and "workpiece carrier" are used analogously. What is always meant here is the combination of a workpiece and its workpiece carrier, which pass through the production system together from the transfer station to the unloading station. When entering the area of the WSs, the workpiece carrier is transported to the first transfer station. Like all other conveyors and transfer units, the first transfer unit has the method

M_{av} assigned as input control. According to the examined movement profile with velocity and acceleration, the appropriate values from the tables $T_{acceleration}$ and $T_{velocity}$ are set for the transfer unit or conveyor and result from physics simulation as described in the following subsections. When leaving the transfer unit, $M_{Destination}$ is called up, which is decisive for the processing of the work plan and the control of the workpiece carriers by the production system. This method can be seen as the most important control method of the present simulation model and in this case, it implements the decentralized control of the conveyors by the production system in the simulation model. The control system integrated with each conveyor and transfer unit consists of two parts: In the first part, the next processing station for the present workpiece carrier is determined by means of several table comparisons. In the second part the transfer to the next conveyor is carried out according to the destination. As shown in the flow chart in Figure 8 , the first part checks whether any work steps are still pending. If this is not the case, the conveyor C_{End} is selected as the next destination and the workpiece carrier moves to the sink. Otherwise, the next work step and the corresponding processing station are selected by comparing several table values. Now all possible processing stations are checked for their assignment. The station first recognized as free is selected as destination and entered in the table $T_{NextDestination}$. If all stations are occupied, the first possible station is again selected as destination and waiting times are accepted.

The second part of the method $M_{Destination}$ is based on three binary questions:

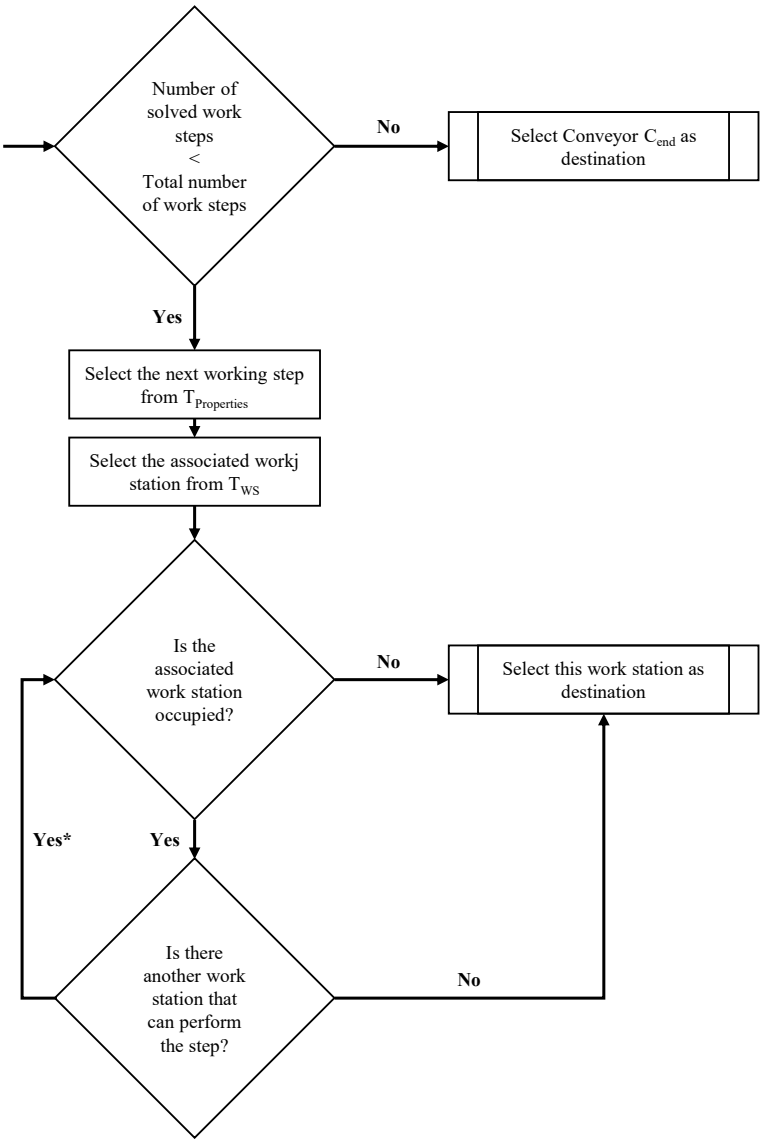
1. On which module is the workpiece carrier currently located?
2. What is the next destination of the workpiece carrier?
3. How must the workpiece carrier be transferred in order to reach its destination by the shortest route?

To implement this method, corresponding work plans are necessary, which are stored for each component in the table $T_{Properties}$. Table 3 shows an example of the working plan for a housing workpiece. Via T_{WS}, the corresponding individual stations are assigned to the work steps. For the component in question, the milling step must first be carried out, which can be performed on the stations “WS 1”, “WS 3” or “WS 8” according to the information in Table 2.

Table 3 Exemplary work plan.

Step number	Process
1	Milling
2	Drilling
3	Deburring
4	Washing
5	Assembly

To select a station, the method $M_{Destination}$ is used and the cycle described in Figure 8 is performed: First, the method checks whether station “WS 1” is free. Assuming that this is the case, the second part of the method decides on which conveyor the workpiece carrier is transferred. Following the principle of the shortest way, the workpiece carrier is transferred to the conveyor “CM1”. If again station “WS 2” or “WS 4” were the next destination, the workpiece carrier would be transferred to conveyor “CM13”. Via the conveyors (where again M_{av} selects the respective acceleration-velocity constellation), the transfer units and the upstream buffer, the workpiece is brought to the corresponding WS. There, in addition to processing, the current individual station is entered in T_{Count} via M_{Count} as the input control of the individual station. This table is used to enable a comparison between the



* This cycle is performed iteratively, until all possible work stations are checked. If all stations are occupied, the cycle restarts at the first possbible station

Figure 8 Flow chart of $M_{Destination}$.

number of steps already completed and the total number of steps in $M_{Destination}$. The method $M_{OutputWS}$, which is integrated into the individual stations as output control, removes the current destination from $T_{NextDestination}$ and transfers the workpiece carrier to the next conveyor. This sequence of operations is repeated until all steps of a work plan are completed and the workpiece carrier is directed to the conveyor C_{End}. Via the input control of this conveyor, the number of workpieces in the system n_{WP} is reduced by 1 with M_{nWP}. The output control $M_{EvaluationWS}$ enters the key figures required for the evaluation in relation to the utilization of the WSs and conveyors in the tables $T_{EvaluationWS}$ or in $T_{EvaluationConveyors}$ as soon as the processing of all 200 workpieces has been completed. Afterwards, the finished workpiece is taken from the workpiece carrier via the unloading station and the downstream

buffer. Both are conveyed to their respective sinks using the M_{Buffer} method. The method $M_{MEvaluation\_throughput}$ computes and enters the key resulting figures related to the actual workpiece in the table $T_{Evaluation\_throughput}$. The discrete-event simulation model performs the operation of the production system. As described above, this requires product-individual acceleration values that are derived with a physics simulation model which is described in the next Subsection.

4.3 Physics simulation model

In order to determine optimal parameters, each material flow process needs to be simulated using physics simulation. This requires physical models that consist of the respective material handling systems as well as of the transported work pieces. Regarding the material handling systems, pallet-type conveyors were used. These conveyors use workpiece pallets that are conveyed on rails and are used to carry the workpieces. To determine the physical behavior of components during transport, it is sufficient to model just one instance of the conveyor, as all conveyors in the production system are alike. Therefore, the conveyor was modeled in pyBullet including the length of the conveyor track and the area of the load bed. Furthermore, a control module was implemented that is able to perform material handling processes with specific parameters of "a" and "v_c" according to the kinematics of Figure 5. In addition, physics simulation models of the produced components (workpieces) were created, including attributes like mass, inertia or friction. For each manufactured component, the ideal transport parameters are defined. For this purpose, the previously described and validated physics simulation model was adapted [16]. Figure 9 shows an exemplary workpiece that is being transported on the conveyor model. Each part that is being produced and transported in the production system was simulated within the physics simulation environment.

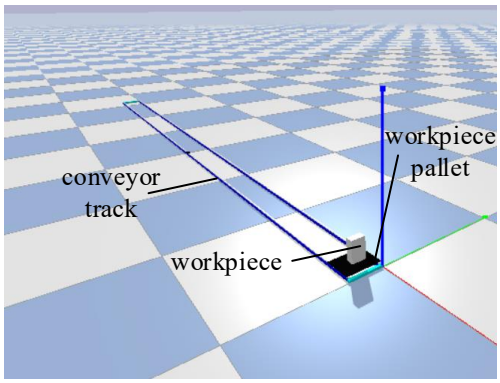


Figure 9 Physics simulation model of the conveyor.

In order to determine the optimal acceleration for each component, a predictive rule-based approach as described in [19] was used (see Figure 10). Using physics simulation, the effect of different acceleration values on the stability of the workpiece is investigated. If the simulation suggests that this stability cannot be maintained for the respective configuration (e.g. the workpiece tips over due to acceleration), this consequence is considered a disturbance and the selected acceleration value cannot be used for the real process. Determining the chosen acceleration is achieved using an iterative approach. For this purpose, a list of different acceleration values from 0.5 to 10.0 m/s² in steps of 0.5 m/s² is generated. For each acceleration value, a simulation run is performed, evaluating the effects on the stability of the transported workpiece. All accelerations that resulted in a disturbance are neglected. Within

the acceleration values that did not cause any disturbance, the highest value is selected as the acceleration for the respective component. This resulted in a list of components and respective maximum acceleration values. The contents of this list are given in Table 4 and serve as input for the discrete-event simulation in the subsequent Section.

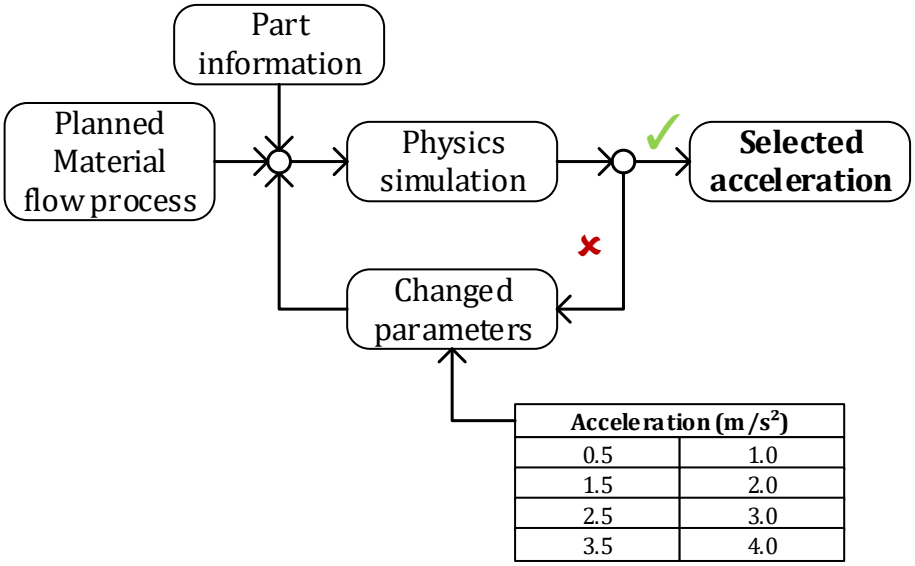


Figure 10 Logical sequence for determining suitable acceleration values.

Table 4 List of simulated components (Id) and the respective identified maximum acceleration value in m/s².

Id	a_{ind}	Id	a_{ind}	Id	a_{ind}	Id	a_{ind}	Id	a_{ind}
Block01	2.0	Gear01	4.5	Housing01	5.5	Plate01	8.5	Shaft01	2.5
Block02	2.5	Gear02	7.5	Housing02	2.5	Plate02	4.5	Shaft02	2.5
Block03	2.5	Gear03	5.5	Housing03	6.5	Plate03	7.5	Shaft03	2.5
Block04	2.5	Gear04	8.5	Housing04	6.5	Plate04	3.5	Shaft04	2.0
Block05	2.0	Gear05	4.5	Housing05	8.5	Plate05	7.5	Shaft05	2.0
Block06	4.0	Gear06	8.5	Housing06	3.5	Plate06	4.5	Shaft06	7.5
Block07	5.0	Gear07	3.5	Housing07	4.5			Shaft07	5.5
Block08	3.0	Gear08	5.5	Housing08	3.5			Shaft08	2.0
Block09	2.5	Gear09	5.5	Housing09	3.5			Shaft09	2.5
Block10	2.5	Gear10	5.5	Housing10	8.5			Shaft10	2.0
		Gear11	7.5					Shaft11	4.5
								Shaft12	5.5
								Shaft13	7.5

4.4 Connection of the models

As described before, the discrete-event simulation model performs production tasks that are performed on various WSs. The transports between those WSs are performed via conveyors. Based on a table that contains physical properties of the 50 produced components, physics simulations are performed and the acceleration parameters are derived as described in Section

4.3. The component-individual physics simulation results shown in Table 4 are exported to an excel file, which can be imported into SIEMENS Plant Simulation. Here, the simulation-based values for acceleration (a_{ind}) are entered in the $T_{acceleration}$ table and are used for the discrete-event simulation.

4.5 Simulation run and results

In order to analyze the effect of acceleration and velocity, 16 scenarios were considered, that are all characterized by a specific acceleration-velocity-pairing. Besides considering the parameters that resulted from the physics simulation (Scenarios 13-16, values from Table 4), fixed accelerations from 1 to 3 m/s² were also incorporated (Scenarios 1-12). All scenarios are summarized in the following Table 5.

Table 5 Simulation scenarios.

		Acceleration in m/s²			
		1	**2**	**3**	**Individual acceleration**
Velocity in m/s	**0.5**	1	5	9	13
	1.0	2	6	10	14
	1.5	3	7	11	15
	2.0	4	8	12	16

For all scenarios, the respective values for acceleration and velocity were passed to the respective tables to be processed within the discrete-event simulation model. Within each scenario, the throughput times of all 200 jobs are analyzed and the mean throughput time is calculated. In order to reduce the impact of random effects, a basic principle in discrete-event simulation is that an increase in the number of simulation runs and a longer simulation duration leads to improved quality of results. This is due to the fact that with a larger sample size the confidence interval becomes smaller and thus the inaccuracy about the exact location of the target value is reduced [42]. A commonly used number of simulation runs is 10 [25]. In this analysis, this value is doubled and therefore 20 simulation runs are performed per scenario. In order to consider random influences on one hand and to make the results comparable on the other hand, the random seed starting value in Plant Simulation, which starts with 1, is increased by one with each repetition. Therefore, randomized orders of incoming jobs are considered in the 20 runs per scenario. Afterwards, the average throughput and utilization values from the 20 runs per scenario are calculated. Figure 11a) shows the resulting throughput times for the 16 scenarios, clustered by values for same acceleration.

It becomes clear that with constant acceleration, increasing the conveyor velocity leads to a significant reduction in the throughput time. The influence of acceleration becomes clearer in Figure 11b), where the diagram shows the throughput time, clustered by values of same velocity. It can be seen in this Figure that in a class with constant velocity, the average throughput time decreases when the acceleration is increased. Comparing the four velocity clusters with each other, it becomes clear that such a decrease compared to the slowest acceleration scenario of a class (scenario 1, 2, 3 or 4) becomes larger with increasing velocity. Consequently, in the cluster $v_c = 2$ m/s, the difference between slowest acceleration and component-specific acceleration (see Table 4) is significantly greater than in the cluster $v_c = 0.5$ m/s. Besides the comparison between scenario 1 and 5, all relevant differences can be seen as significant (see Appendix A). Within a cluster of equal velocity, the scenario with

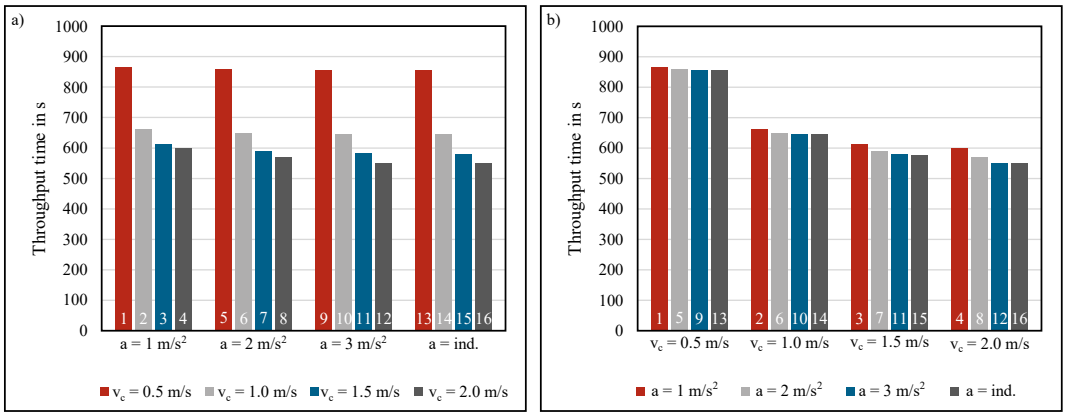


Figure 11 Resulting throughput times; a) clustered by acceleration; b) clustered by velocity.

component-specific acceleration using the physics simulation leads to the shortest processing time in each case. Overall, the selection of the motion profile in scenario 16 (v_c = 2 m/s and a = component-specific) leads to the shortest processing time of all scenarios. It can be concluded, that the influence of high accelerations becomes especially important with high transport velocities. In addition to throughput, capacity utilization is another commonly chosen performance indicator in a production system. This quantity describes the ratio of actual to maximum possible utilization of a resource [32]. In this context, the capacity utilization of the conveyors with regards to velocity and acceleration were analyzed. The results are shown in Figure 12.

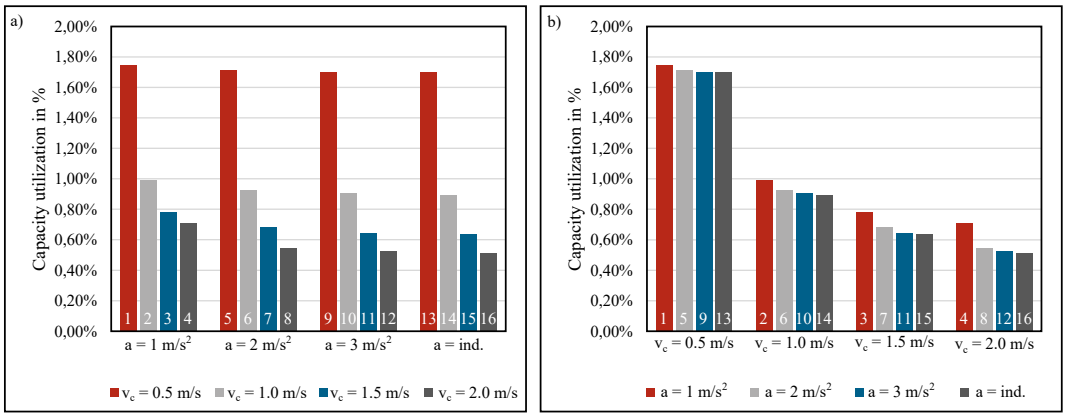


Figure 12 Resulting capacity utilization of conveyors: a) clustered by acceleration; b) clustered by velocity.

The choice of the acceleration-velocity constellation has a significant effect on the conveyor utilization. Figure 12a) shows significant changes in the utilization of the conveyors with increasing velocity. As the bar graphs in this Figure show, each increase in velocity leads to a reduced utilization of the conveyors, which means that they have higher capacities for further production orders. Within a cluster of constant acceleration, in particular the doubling of the velocity from v_c = 0.5 m/s to v_c = 1 m/s leads to a strong effect on the capacity utilization, which is almost reduced by half. A similar effect can be observed when doubling the velocity from v_c = 1 m/s to v_c = 2 m/s, which emphasizes the positive effect

of a possible technical realization of increased conveyor velocities. If the utilization of the conveyors is clustered by variable accelerations at constant velocities, an influence of the acceleration is observed (see Figure 12b)). Every increase in acceleration results in a lower utilization of the conveyors. Although the differences are not as great as between the velocity variations, a clear tendency can be seen here as well. Apart from the velocity class $v_c = 0.5$ m/s, the correlation can be described as significant. In this case, the effects of acceleration on the utilization of the conveyors become stronger with increasing velocity. As a consequence, in the case of a component-specific acceleration (see Table 4) and a velocity of $v_c = 2$ m/s (scenario 16), the lowest utilization is recorded. All differences regarding capacity utilization can be seen as significant, excluding the differences in the $v_c = 0.5$ m/s cluster (see Appendix A). In conclusion, the results indicate that the choice of conveyor velocity has an influence on the throughput time and its utilization. With increasing velocity, the throughput time can be reduced and the utilization of the conveyors can be lowered. Also, the influence of acceleration can be recognized. With increasing acceleration, both the throughput time and the utilization of the conveyors can be reduced. In general, the effect of acceleration on both target variables is less than with increasing velocity, but is clearly visible and significant in the figures. In this respect, the alignment of the acceleration to workpiece-specific conditions in particular leads to the lowest values in terms of throughput time and conveyor utilization compared to other acceleration variants. This indicates that using physics simulation to determine component-specific accelerations can be beneficial to the operation of a production system.

4.6 Discussion of results

The results indicate that the velocity of the material flow strongly influences throughput time in a production system. It becomes obvious that the constant velocity v_c is the most important value that has the largest impact on throughput time. In comparison, the effect of the accelerations which are calculated with physics simulations seems small. However, the results show that the effect of high accelerations becomes stronger with higher velocities. This seems reasonable, since high conveying speeds can only be reached quickly with an adequate acceleration. This is especially true on short transport distances. Therefore, achieving shorter material flow times benefits from high velocities but is also enabled by high accelerations.

It can therefore be concluded that physics simulation can provide a substantial improvement for material flows in future production systems: Using adequate models of material handling systems and transported components, the potentials of faster material flows can be exploited, resulting in better production system performance.

In order to reduce the modeling efforts, all components were assumed to be physically complete from the beginning of their production. This was done to reduce the physical modeling efforts, since otherwise, a physical model of each manufacturing sub-step (e.g. raw parts and semi-finished components) of all 50 products would have to be created. This is a simplification, since in a real scenario, the physical attributes of a product may change in the course of a manufacturing process, for instance through removed material and hence changed mass distribution. Overcoming this simplification would have included to generate three-dimensional (e.g. CAD) data about the geometry of every sub-step, along with the respective mass distribution (e.g. by calculating the removed mass during a milling process) and friction coefficient. While this can be performed manually by assessing every manufacturing step and the effects on the physical properties of a component, future research could investigate the utilization of respective simulation or calculation methods (e.g. material removal simulations) that can derive physical models of sub-steps automatically. Further aspects can be added to

the simulation study to represent the complexity that is incorporated in industrial settings. The kinematical characteristic (see Figure 5) might be extracted from real conveyors, which might show non-linear acceleration. In addition, physical obstacles like curves or slopes that are often found in real factories can be added to the conveyors. Additionally, machine failures or maintenance times can be added to the discrete-event simulation. This also affects conveyors which might need to be repaired as well, which results in a limited availability.

Furthermore, exploiting the operational benefits of faster material flows in real production systems is connected to certain challenges. It is likely, that higher velocities and accelerations lead to higher energy consumptions of the material handling systems which need to be considered. Furthermore, compared to the slow but steady operation of most current material handling systems, the alternating operation that is connected to component-specific accelerations needs to be incorporated during the design of these systems (e.g. drives and bearings).

5 Summary and outlook

Material flows that are performed by material handling systems account for a significant percentage of the manufacturing cost of a product. For preventing disturbances as a result of strong accelerations and inertia forces, material flows are usually performed at comparably low velocities and accelerations. As a result, non-value-adding transport times increase throughput times and block conveyors. Despite the safety advantages, potentials to improve the operation of a production system by faster material flows are mostly unexploited. Physics simulation using physics engines has indicated the potential of determining component-individual accelerations. However, the effects on the performance of a production system have not been investigated in the past.

In order analyze these effects, this paper analyzed the influence of velocity and acceleration on throughput time and capacity utilization of conveyors within a production system. For this purpose, a discrete-event simulation was combined with a physics simulation. The results indicate that throughput times can be significantly reduced by choosing faster velocities for material handling systems. The influence of accelerations was found to be smaller, however it became increasingly greater with higher velocities. This means that high accelerations serve as an enabler to allow for performance optimizations through high velocities. A consideration of component-individual accelerations therefore seems useful when operating material handling systems with high velocities.

Despite the potential benefits through faster material flows, the optimization of routing and scheduling still plays a crucial role in production system operation. It seems promising to further investigate the described coupling of discrete-event and physics simulation with the goal of optimizing routing and scheduling along with ensuring the physical feasibility of the material flows. Further research also needs to evaluate the barriers of implementing the investigated component-individual acceleration control in a factory environment. Furthermore, sub-steps of manufactured components can be included within the physics simulation (see Section 4.6).

References

1 Bugra Alkan and Robert Harrison. A virtual engineering based approach to verify structural complexity of component-based automation systems in early design phase. *Journal of Manufacturing Systems*, 53:18–31, 2019. doi:10.1016/j.jmsy.2019.09.001.

2 Dieter Arnold. *Intralogistik: Potentiale, Perspektiven, Prognosen.* VDI-Buch. Springer-Verlag GmbH, Berlin Heidelberg, 2006. doi:10.1007/978-3-540-29658-4.

3 Dieter Arnold and Kai Furmans. *Materialfluss in Logistiksystemen.* VDI-Buch. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 6., erweiterte aufl. edition, 2009. doi:10.1007/978-3-642-01405-5.

4 Jan Bender, Kenny Erleben, and Jeff Trinkle. Interactive simulation of rigid body dynamics in computer graphics. *Computer Graphics Forum*, 33(1):246–270, 2014. doi:10.1111/cgf.12272.

5 BOSCH Rexroth. Transfer system ts 2plus. URL: https://www.boschrexroth.com/en/xc/products/product-groups/assembly-technology/topics/transfer-systems/transfer-system-ts-2plus/index.

6 Manfred Broy. *Cyber-Physical Systems.* Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. doi:10.1007/978-3-642-14901-6.

7 Erwin Coumans and Yunfei Bai. Pybullet: A python module for physics simulation for games, robotics and machine learning, 2019. URL: http://pybullet.org.

8 Klaus Dröder, Franz Dietrich, Christian Löchte, and Jürgen Hesselbach. Model based design of process-specific handling tools for workpieces with many variants in shape and material. *CIRP Annals*, 65(1):53–56, 2016. doi:10.1016/j.cirp.2016.04.109.

9 David H. Eberly. *Game physics.* Interactive 3D technology series. Morgan Kaufmann/Elsevier, Amsterdam, 2. ed. edition, 2010. URL: http://www.eblib.com/patron/FullRecord.aspx?p=648814.

10 H. A. ElMaraghy and M. Manns. Transition of interarrival time patterns between automated and manual configurations of assembly systems. *Journal of Manufacturing Systems*, 26(1):1–11, 2007. doi:10.1016/j.jmsy.2008.01.001.

11 Hoda ElMaraghy and Alessandra Caggiano. Flexible manufacturing system. In Luc Laperrière and Gunther Reinhart, editors, *CIRP encyclopedia of production engineering*, Springer reference, pages 524–530. Springer, Berlin, 2014. doi:10.1007/978-3-642-20617-7_6554.

12 Behzad Esmaeilian, Sara Behdad, and Ben Wang. The evolution and future of manufacturing: A review. *Journal of Manufacturing Systems*, 39:79–100, 2016. doi:10.1016/j.jmsy.2016.03.001.

13 Marc-André Filz, Johann Gerberding, Christoph Herrmann, and Sebastian Thiede. Analyzing different material supply strategies in matrix-structured manufacturing systems. *Procedia CIRP*, 81:1004–1009, 2019. doi:10.1016/j.procir.2019.03.242.

14 Marc-André Filz, Christoph Herrmann, and Sebastian Thiede. Simulation-based data analysis to support the planning of flexible manufacturing systems. *SNE Simulation Notes Europe*, 30(4):131–137, 2020. doi:10.11128/sne.30.tn.10531.

15 Moritz Glatt and Jan C. Aurich. Physiksimulation cyber-physischer produktionssysteme: Planung und steuerung cyber-physischer produktionssysteme durch physikalische simulation. *wt Werkstatttechnik online*, 108(4), 2018.

16 Moritz Glatt and Jan C. Aurich. Physical modeling of material flows in cyber-physical production systems. *Procedia Manufacturing*, 28:10–17, 2019. doi:10.1016/j.promfg.2018.12.003.

17 Moritz Glatt, Georg Kasakow, and Jan C. Aurich. Combining physical simulation and discrete-event material flow simulation. *Procedia CIRP*, 72:420–425, 2018. doi:10.1016/j.procir.2018.03.054.

18 Moritz Glatt, Daniel Kull, Bahram Ravani, and Jan C. Aurich. Validation of a physics engine for the simulation of material flows in cyber-physical production systems. *Procedia CIRP*, 81:494–499, 2019. doi:10.1016/j.procir.2019.03.125.

19 Moritz Glatt, Chantal Sinnwell, Li Yi, Sean Donohoe, Bahram Ravani, and Jan C. Aurich. Modeling and implementation of a digital twin of material flows based on physics simulation. *Journal of Manufacturing Systems*, 2020. doi:10.1016/j.jmsy.2020.04.015.

20 Daniel Hofmann and Gunther Reinhart. Raising accuracy in physically based simulations through scaling equations. *Journal of Computing and Information Science in Engineering*, 13(4):435, 2013. doi:10.1115/1.4025590.

21 S. Hoher, P. Schindler, S. Göttlich, V. Schleper, and S. Röck. System dynamic models and real-time simulation of complex material flow systems. In Hoda A. ElMaraghy, editor, *Enabling Manufacturing Competitiveness and Economic Sustainability*, pages 316–321. Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2012. doi:10.1007/978-3-642-23860-4_52.

22 Danny J. Johnson. A framework for reducing manufacturing throughput time. *Journal of Manufacturing Systems*, 22(4):283–298, 2003. doi:10.1016/S0278-6125(03)80009-2.

23 Laurie King Rogers. Automatic guided vehicles basics. *Modern Materials Handling*, pages 36–39, 2011.

24 Ravindra Kumar, Abid Haleem, Suresh K. Garg, and Rajesh K. Singh. Automated guided vehicle configurations in flexible manufacturing systems: a comparative study. *International Journal of Industrial and Systems Engineering*, 21(2):207, 2015. doi:10.1504/IJISE.2015.071510.

25 Averill M. Law. *Simulation modeling and analysis*. McGraw-Hill series in industrial engineering and management science. McGraw-Hill Education, New York, NY, 5. ed. edition, 2015.

26 Ian Millington. *Game Physics Engine Development: How to Build a Robust Commercial-Grade Physics Engine for your Game*. CRC Press, London, 2nd ed. edition, 2010. URL: http://gbv.eblib.com/patron/FullRecord.aspx?p=4205386.

27 L. Monostori, B. Kádár, T. Bauernhansl, S. Kondoh, S. Kumara, G. Reinhart, O. Sauer, G. Schuh, W. Sihn, and K. Ueda. Cyber-physical systems in manufacturing. *CIRP Annals - Manufacturing Technology*, 65(2):621–641, 2016. doi:10.1016/j.cirp.2016.06.005.

28 Dimitris Mourtzis. Simulation in the design and operation of manufacturing systems: state of the art and new trends. *International Journal of Production Research*, 58(7):1927–1949, 2020. doi:10.1080/00207543.2019.1636321.

29 Dimitris Mourtzis, Nikolaos Papakostas, Dimitris Mavrikios, Sotiris Makris, and Kosmas Alexopoulos. The role of simulation in digital manufacturing: Applications and outlook. *International Journal of Computer Integrated Manufacturing*, 28(1):3–24, 2014. doi:10.1080/0951192X.2013.800234.

30 Ashkan Negahban and Jeffrey S. Smith. Simulation for manufacturing system design and operation: Literature review and analysis. *Journal of Manufacturing Systems*, 33(2):241–261, 2014. doi:10.1016/j.jmsy.2013.12.007.

31 Luis Rabelo, Magdy Helal, Albert Jones, and Hyeung-Sik Min. Enterprise simulation: a hybrid system approach. *International Journal of Computer Integrated Manufacturing*, 18(6):498–508, 2005. doi:10.1080/09511920400030138.

32 REFA. Auslastung, 2020. URL: https://refa-consulting.de/refa-lexikon/a/auslastung.

33 Gunther Reinhart and Frédéric-Felix Lacour. Physically based virtual commissioning of material flow intensive manufacturing plants. In Michael F. Zäh, editor, *3rd International Conference on Changeable, Agile, Reconfigurable and Virtual Production (CARV 2009)*, pages 377–386, München, 2009. Herbert Utz Verlag.

34 Frank Riley. *The Electronics Assembly Handbook*. Springer, Berlin and Heidelberg, 1988. doi:10.1007/978-3-662-13161-9.

35 SIEMENS. Use plant simulation and throughput optimization to improve manufacturing performance, 2020. URL: https://www.plm.automation.siemens.com/global/de/products/manufacturing-planning/plant-simulation-throughput-optimization.html.

36 Stein Automation GmbH & Co. KG. Modular und vernetzt in die montage der zukunft. *Automationspraxis*, 2018. URL: https://www.stein-automation.de/wp-content/uploads/Montage-der-Zukunft_Titelstory_automationspraxis-7_2018.pdf.

37 Matthew P. Stephens and Fred E. Meyers. *Manufacturing facilities design and material handling*. Purdue University Press, West Lafayette, Indiana, fifth edition edition, 2013.

URL: http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=582967.

38 George Swartz. *Forklift safety: A practical guide to preventing powered industrial truck incidents and injuries*. Government Institutes, Rockville, MD, 1999.

39 Insup Um, Hyeonjae Cheon, and Hongchul Lee. The simulation design and analysis of a flexible manufacturing system with automated guided vehicle system. *Journal of Manufacturing Systems*, 28(4):115–122, 2009. doi:10.1016/j.jmsy.2010.06.001.

40 Birgit Vogel-Heuser, Jay Lee, and Paulo Leitão. Agents enabling cyber-physical production systems. *at - Automatisierungstechnik*, 63(10), 2015. doi:10.1515/auto-2014-1153.

41 Shiyong Wang, Jiafu Wan, Di Li, and Chunhua Zhang. Implementing smart factory of industrie 4.0: An outlook. *International Journal of Distributed Sensor Networks*, 12(1):3159805, 2016. doi:10.1155/2016/3159805.

42 Sigrid Wenzel. *Qualitätskriterien für die Simulation in Produktion und Logistik: Planung und Durchführung von Simulationsstudien*. VDI-Buch. Springer, Berlin, 2008. URL: http://deposit.dnb.de/cgi-bin/dokserv?id=2814919&prov=M&dok_var=1&dok_ext=htm.

43 H.-P. Wiendahl, H. A. ElMaraghy, P. Nyhuis, M. F. Zäh, H.-H. Wiendahl, N. Duffie, and M. Brieke. Changeable manufacturing - classification, design and operation. *CIRP Annals*, 56(2):783–809, 2007. doi:10.1016/j.cirp.2007.10.003.

44 Michael F. Zäh, Michael Spitzweg, and Frédéric-Felix Lacour. Application of a physical model for the simulation of the material flow of a manufacturing plant (einsatz eines physikmodells zur simulation des materialflusses einer produktionsanlage). *it - Information Technology*, 50(3):192–198, 2008. doi:10.1524/itit.2008.0483.

A Significance analysis

In order to determine significance, Welch's-t-test is applied (see [25]). The confidence interval (CI) between two results was calculated according to Equation 1:

$$CI = \overline{X_1} - \overline{X_2} \pm t_{f,1-\frac{\alpha}{2}} \cdot \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \tag{1}$$

Here, $\overline{X_1}$ and $\overline{X_2}$ are the mean values of the respective scenarios regarding throughput time and utilization. s denotes the standard deviation, while n stands for the number of samples (4000). The two-tailed t-value was calculated with the degrees of freedom f according to Equation 2. Tables 6 and 7 show the significance analysis results. The difference is seen as significant, if the CI does not include 0.

$$f = \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{(n_1-1)} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{(n_2-1)}} \tag{2}$$

Table 6 Significance analysis for throughput time.

Compared scenarios	Lower endpoint welch interval	Upper endpoint welch interval	Significant difference?
1 vs. 2	193.49	212.27	yes
1 vs. 3	243.04	261.68	yes
1 vs. 4	257.06	276.75	yes
5 vs. 6	200.21	218.68	yes
5 vs. 7	258.63	277.00	yes
5 vs. 8	278.37	298.00	yes
9 vs. 10	198.27	217.11	yes
9 vs. 11	263.73	281.94	yes
9 vs. 12	294.85	313.40	yes
13 vs. 14	200.56	219.50	yes
13 vs. 15	267.38	285.44	yes
13 vs. 16	295.75	314.06	yes
1 vs. 5	-3.06	17.87	no
1 vs. 9	0.63	21.48	yes
1 vs. 13	0.54	21.45	yes
2 vs. 6	5.97	21.97	yes
2 vs. 10	7.60	24.12	yes
2 vs. 14	9.85	26.43	yes
3 vs. 7	15.00	30.71	yes
3 vs. 11	23.71	39.34	yes
3 vs. 15	27.35	42.73	yes
4 vs. 8	19.53	37.84	yes
4 vs. 12	39.66	56.89	yes
4 vs. 16	40.53	57.46	yes

Table 7 Significance analysis for capacity utilization.

Compared scenarios	Lower endpoint welch interval	Upper endpoint welch interval	Significant difference?
1 vs. 2	0.0067	0.0084	yes
1 vs. 3	0.0089	0.0104	yes
1 vs. 4	0.0096	0.0112	yes
5 vs. 6	0.0070	0.0087	yes
5 vs. 7	0.0095	0.0111	yes
5 vs. 8	0.0109	0.0124	yes
9 vs. 10	0.0071	0.0088	yes
9 vs. 11	0.0098	0.0113	yes
9 vs. 12	0.0110	0.0125	yes
13 vs. 14	0.0073	0.0089	yes
13 vs. 15	0.0099	0.0114	yes
13 vs. 16	0.0111	0.0126	yes
1 vs. 5	-0.0006	0.0013	no
1 vs. 9	-0.0005	0.0014	no
1 vs. 13	-0.0005	0.0014	no
2 vs. 6	0.0001	0.0012	yes
2 vs. 10	0.0003	0.0014	yes
2 vs. 14	0.0004	0.0015	yes
3 vs. 7	0.0006	0.0014	yes
3 vs. 11	0.0010	0.0018	yes
3 vs. 15	0.0011	0.0019	yes
4 vs. 8	0.0013	0.0020	yes
4 vs. 12	0.0014	0.0021	yes
4 vs. 16	0.0016	0.0023	yes

Physical Modeling of Process Forces in Grinding

Praveen Sridhar ✉
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Daniel Mannherz
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Raphael Bilz
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Kristin M. de Payrebrune
Institute of Computational Physics in Engineering, Technische Universität Kaiserslautern, Germany

Mahesh R.G. Prasad
ICAMS, Ruhr-Universität Bochum, Germany

Juan Manuel Rodríguez Prieto
Mechanical Engineering Department, Universidad EAFIT, Medellín, Colombia

Abstract

This paper deals with material removal mechanisms in grinding by considering single grit-workpiece interactions. Individual investigations were performed both experimentally and using finite element simulations. Firstly, a comparison between the Johnson-Cooke material model and a Crystal Plasticity finite element method was performed with the help of micro-indentation experiments. Here the research question was answered if an anisotropic material model better describe the grinding process and process forces compared to an isotropic material model. Secondly, four discretization approaches were employed: pure Lagrangian (LAG), Arbitrary Lagrange Eulerian (ALE), Particle Finite Element Method (PFEM), and Smooth Particle Hydrodynamics (SPH), to simulate a micro-cutting operation of A2024 T351 aluminium. This study aims to compare the conventional approaches (LAG and ALE) to newer approaches (PFEM and SPH). The orthogonal cutting models were benchmarked against a micro-cutting experiment presented in literature, by comparing the obtained cutting and passive forces. The study was then extended to negative rake angles to study the effect on the discretization approaches for grinding. Thirdly, scratch experiments were investigated for a brittle material sodalime glass and A2024 T351 aluminium. Effects of the linear speed of the device, depth of cut, and conical tool angle were analyzed and tendencies are built. Finally, a realistic simulation of the manufacturing process of a grinding wheel was developed, starting with the raw material, compression, sintering, and dressing until the final grinding surface. As a result of the simulations, virtual grinding wheel topographies can be visualized and analyzed with regard to the output variables from grinding wheels such as bonding strength and static grain count. The individual research studies help in understanding the material removal mechanisms in a single grit scratch process as well as in the understanding of the overall grinding wheel topography. This in turn helps in the developing an overall physical force model for scratching/grinding to predict mechanical output parameters and hence reduce the need for experimentation.

2012 ACM Subject Classification Computing methodologies → Modeling and simulation

Keywords and phrases grinding, single grit approach, finite element method, smooth particle hydrodynamics, particle finite element method, scratch experiments, virtual grinding wheel model

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.16

Funding This research work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) Project ID– 252408385 – IRTG 2057 and Project-ID 172116086 – SFB 926.

Acknowledgements The authors would like to thank Matthias W. Klein, Marek Smaga and Tilmann Beck from the Institute of Material Sciences and Engineering at Technical Universität Kaiserslautern for their collaboration and for providing us with the experimental data used in section 2.

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 16; pp. 16:1–16:20

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Grinding and Single Grit Scratch Approach

Grinding is a form of material removal applied to many metals and ceramics, especially hard-to-machine materials, to obtain smooth surfaces on products within a desired tight tolerance. Grinding mechanics and material removal mechanisms are complex and the processes are still not fully understood. In the manufacturing industry, almost 65 % of machining companies use the grinding process [5] and a significant portion of production cost is due to grinding or related abrasive processes. However, some experimental observations are difficult to explain theoretically due to the complex nature of the process mechanisms. Also, the determination of some of the mechanical characteristics, e.g., stresses, strains, and temperature distribution across the machined surface are difficult to obtain experimentally and the grinding process analysis using experimental tests could be both costly, owing to the high price of grinding tools, and not generally valid as the results obtained might be valid only for a certain machining condition [17]. Thus, the prediction of mechanical parameters such as forces, strains, stresses, and surface deformations that occur during grinding using computer technology is a valuable tool of getting additional insight into the final quality of machined surface and to reduce costly experiments.
The grinding process is a material removal process using a grinding wheel, which is made up of a large number of randomly oriented abrasive particles. The grinding operation is performed at relatively high speed and the undeformed chip thickness is relatively small particularly in fine grinding operations compared to other machining processes. To measure the grinding performance, the grinding system behavior must be evaluated properly. The grinding system behavior is mainly assessed by considering abrasive geometry, kinematics, mechanics, and energy and material properties[17]. The Overall grinding performance including the finish surface quality is directly related to the performance of the grinding wheel used during the grinding operation. The performance of a precision grinding wheel is usually determined by using the parameters such as grinding force, machining vibration, the temperature in the cutting zone, and workpiece surface roughness.

Material removal mechanisms for grinding can be categorized into two groups: (i) removal mechanisms during grinding wheel-workpiece interaction and here investigations will be at the macroscale, and (ii) removal mechanisms considering single grit-workpiece interaction and investigations will be at the microscale.
The following individual studies were performed under these two groups:

- Parametristaion of a material model employed in the finite elemenet (FE) model is important in the prediction of the process variables in a single grit scratch process. Here a research study is performed to compare an isotropic Johnson-Cooke model with an anisotropic crystal plasticity FEM model with help of micro-indentation tests, to understand the effect of the choice of a material model (ref section 2).
- A FE material removal model is developed to understand the cutting mechanism taking place during a single grit scratch. The model studies in detail the effect of change in the tool geometry (rake angle) on the process forces. Furthermore, the individual models are discretized using different available discretizational approaches, to understand its advantages and drawbacks, as described in section 3.
- Experimental investigations of single grit scratching is performed using shaped tool geometries with high rake angles between -45° to -75° for a brittle material (glass) and a ductile material (aluminium), ref section 4.
- A realistic simulation of the manufacturing process of a grinding wheel, starting with the raw material mixing, compression, sintering, and dressing until the final generation of a

grinding surface is done in section.5. This research studies the effect of different levels of compression, i.e. the percentage of compression and compression force on grinding wheel measurements such as static grain count and bonding forces.

2 Material Modelling and Parametrisation

In a finite element simulation, material modeling and its parametrization is an innate aspect in predicting the material behavior when subjected to different loading conditions. Specifically, grinding is a surface finishing process involving the interaction of multiple abrasive grains onto a workpiece. Interestingly, in this manufacturing process, the crystal grain size of the workpiece material is in the same range as the removal depth. Hence, a fundamental research question arises: *does an anisotropic material model better describe the grinding process and predict the process forces compared to an isotropic material model?* As grinding is a complex material removal process, initially a simple micro-indentation model is chosen to validate and compare two plasticity models. To describe the anisotropic polycrystalline model, the crystal plasticity finite element method (CPFEM) is employed, and to describe the isotropic model, the Johnson-Cooke plasticity model is chosen. In this study, micro indentation experiments were performed on twinning induced plasticity (TWIP) steel and the respective force-displacement curves were obtained. These curves were subsequently used to validate the chosen plasticity models.

2.1 Experimental Investigations

The experimental investigations of the high manganese HSD600 TWIP steel were performed by Klein, et al. [11]. Firstly, the microstructure of HSD600 showing the grain size distribution and grain orientation mapping is obtained by electron backscatter diffraction (EBSD). The average equivalent grain size was found to be 8.6 μm. The data obtained from the microstructural analysis were used as inputs in the development of the CPFEM model. Secondly, cyclic micro-indentation tests were performed on HSD600 using a Vickers indenter. The specimens were subjected to a maximum load of 1 N for a loading cycle of 6 seconds and the applied load was gradually reduced during the unloading cycle of the following 6 seconds. The load-displacement curves obtained from the micro-indentation tests (ref Fig.1) were used as the basis to validate and compare the results from the FEM simulations.

2.2 Simulation Framework

For the isotropic material model, the Johnson-Cooke material model (JC-model) was employed. This computational model for flow stresses includes the effects of strain hardening, strain rate hardening, and thermal softening. The von Mises flow stress σ, is expressed as the product of the three uncoupled terms,

$$\sigma = [A + B(\varepsilon)^n] \cdot \left[1 + C \ln\left(\frac{\dot{\varepsilon}}{\dot{\varepsilon_0}}\right)\right] \cdot \left[1 - \left(\frac{T - T_{room}}{T_{melt} - T_{room}}\right)^m\right]. \tag{1}$$

In here, A is the quasi-static yield stress, B is the modulus of strain hardening, n is the work hardening exponent, C is the strain rate sensitivity and m defines the temperature sensitivity. The material parameters were initially obtained from [11, 26], although parameter optimizations were performed to obtain a better fit to the experimental results. The simulations of the micro-indentation were performed using the commercial FEM program Abaqus/Standard (version 2019) with the optimized JC-model parameters listed in Table 1.

Table 1 Optimized isotropic Johnson-Cook material parameters.

A [MPa]	B [MPa]	C	n	m
619.96	1642.68	0.098	0.29	1.402

For the anisotropic material model, to predict the response in a polycrystalline material, the whole material is approximated by smaller, periodically repeating material elements, commonly referred to as representative volume element (RVE). The RVE was generated based on the average grain size and texture obtained from the EBSD analyses. The crystal plasticity (CP) model employed to reproduce the plastic deformation in the RVE is implemented based on the model described in [22]. The flow rule that expresses the accumulative shear rate at each slip system α is given by

$$\dot{\gamma}_\alpha = \dot{\gamma}_0 \left| \frac{\tau^\alpha}{\hat{\tau}^\alpha} \right|^{p_1} \operatorname{sgn}(\tau^\alpha), \tag{2}$$

where $\dot{\gamma}_0$ is the reference shear rate, p_1 is the inverse of the strain rate sensitivity, τ^α is the resolved shear stress and $\hat{\tau}^\alpha$ is the slip resistance, which describes the hardening behavior of the material. Simulations were performed using Abaqus/Standard through a UMAT (user defined material subroutine) with the following optimized CP parameters in Table 2.

Table 2 Optimized CP parameters for HSD600 TWIP steel used in the CPFEM simulations. The elastic constants were obtained from [18].

C_{11}[GPa]	C_{12}[GPa]	C_{14}[GPa]	p_1	p_2	$\dot{\gamma}_0$ $[s^{-1}]$	$\hat{\tau}_0^\alpha$	$\hat{\tau}_f$[MPa]	h_0[MPa]
174,000	85,000	99,000	13	2.25	0.001	200	900	1600

2.3 Results

Fig.1. shows the force-displacement curves from the micro-indentation experiment of HSD 600 TWIP steel, in comparison to the simulation results of the isotropic JC-model and the anisotropic CPFEM model. On first glance, both material models show a good agreement with the experimental results. Both models closely predict the maximum displacement of 3.6 μm at a maximum load of 1 N. On looking closely at the material response during the loading cycle, both models show a similar response during the elastic domain. As the curves progress towards the plastic domain, there is a divergence of the simulation results compared to the experimental result. The fitting to the experimental data is a little better for the CPFEM model when compared to the JC-model, because the CPFEM model incorporates explicitly the polycrystalline microstructure in contrast to the isotropic JC-model, which does not.

However, the isotropic JC-model is good enough to predict the material behavior of a ductile material and could further be optimized with the help of the CPFEM model by parameter linking. In further studies, EBSD analysis and micro-indentation are planned to be performed on aluminium A2024 T351 and the parameterized isotropic JC-model is used in the scratch simulation to be validate by experiments.

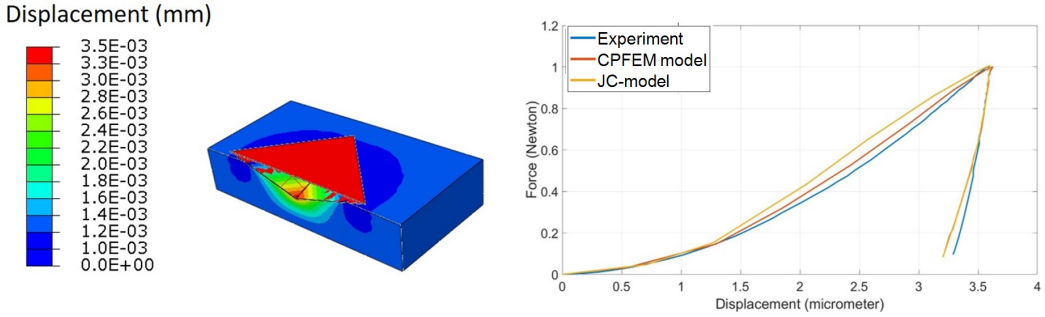


Figure 1 Comparison of force-displacement curves from micro-indentation of HSD 600 TWIP steel between anisotropic CPFEM and isotropic JC-model.

3 Numerical Simulations and 2D Cutting Model

Simulating dynamic machining processes by simplifying the real system is quite commonly used to predict machine behavior and understand the influence of machining parameters on both, tool and workpiece. With increasing computational power it is easier to solve complex problems such as abrasive grit-workpiece interactions [17]. A well-designed simulation model of the grinding wheel could largely reduce the cost of experimentation and successfully determine the grinding process behavior based on the surface finish of the machined part, workpiece characteristics, material removal mechanisms, and tool wear. In addition, simulations are able to provide some extra information like stresses, strains, strain rates, and temperature gradients, which are otherwise extremely difficult to determine by experimental investigations.[17] As grinding is one of the most complex machining processes, there is no universal model to comprehensively predict the machining behavior in terms of the grinding forces, material removal, and surface finish. Hence different numerical approaches are studied in terms of their advantages and drawbacks in modeling grit workpiece interactions.

3.1 Discretization Approaches

In continuum mechanics, the motion of a material point can be described using three algorithms namely the Eulerian, Lagrangian, and Arbitrary Lagrangian-Eulerian methods. In addition, meshless methods like smooth particle hydrodynamics and particle finite element can be employed to describe the material motion.
As part of the comparative study, the two mesh-based methods Lagrangian (LAG) and Arbitrary Lagrangian-Eulerian (ALE) approach, and two particle-based methods Smooth Particle Hydrodynamics (SPH) and Particle Finite Element Method (PFEM) are used to perform cutting simulations of A2024 T351 aluminium. As an outcome of the study, the solution accuracy of the forces, stresses, and temperatures inside the workpiece is compared between the mesh-based methods and particle-based methods for different cutting conditions.

3.1.1 Mesh Based Methods

The Lagrangian (LAG) approach is most often used in solid mechanics problems, where the displacement vector tracks the individual material points. However, its inability to follow large deformation can lead to simulation failures. Possible solutions, as implemented by [15], are either element deletion, which leads to a reduction of mass, or adaptive remeshing. However, in adaptive remeshing, the diffusion of internal variables while mapping from mesh to mesh may invalidate the numerical results of the simulation.

In the Arbitrary Lagrange Eulerian (ALE) approach, the mesh is neither constrained to a fixed space (Eulerian approach) nor does it move along with material points (Lagrangian approach). The ALE approach helps to maintain the mesh topology and avoid iterative meshing. A smoothening algorithm is applied to relocate the nodal positions of the mesh to avoid problems of mesh distortion and hour glassing, as in a similar implementation of [16]. Although the ALE method is computationally effective, the accuracy is still limited to the underlying mesh and the choice of the smoothening algorithm. For the cutting simulations presented in this work, both discretization approaches are used. The workpiece and tool were both modeled in 2D and discretized with temperature coupled quadrilateral continuum elements. The thickness of both bodies was set to 1 mm, similar to the feed width of cut a_p of experiments conducted in [1].

3.1.2 Particle Based Methods

The Smooth Particle Hydrodynamics (SPH) method differs from the above-mentioned techniques since it is categorized as a meshless method. The material domain is discretized by spheres, which can have a strong connection in the case of solids or a weak connection in the case of liquids. Due to the weak bonding of the adjacent particles, a failure criterion is not necessary, since the material separation occurs due to the loss of cohesion between the particles. In the SPH simulation, the cutting tool is modeled as a rigid body, which means that the heat transfer between tool and workpiece cannot be simulated. However, material parameters are provided for the contact analysis.
The Particle Finite Element Method (PFEM) was first introduced by [7] for fluid-structure interaction problems. It combines the benefits of continuum based methods and discrete modeling techniques, therefore it is well suited for problems with large deformations and large configuration changes. In contrast to classical finite element approximations, in PFEM the deformation of the body is calculated by the underlying set of particles. At the end of each time step, a mesh is built upon the particles with their centerpoints as nodes. For this reason, a fast and robust algorithm is essential to build a new mesh. In this work, the *Delaunay Tessellation* method is chosen to connect all particles at the current time step position and creating the new mesh. The particles of the PFEM simulation are spheres and the generated mesh is an unstructured mesh with 2D triangular elements. The diameter of the particles and the thickness of the mesh elements were again set to a feed width of cut $a_p = 1$ mm. The resulting mesh not only works as support where the differential equations are integrated, but is also used to identify contacts and to track free surfaces. The track of the surface is accomplished with the help of a technique called alpha-shape as implemented by [7]. An application of PFEM to the numerical modeling of metal cutting processes can be found in [20].

3.2 Material Model and Damage Model

In metal cutting large strains, large strain rates and high temperatures occur in the primary and secondary shear zones. A material model that describes the material behavior under these conditions is the Johnson-Cook material model, with its mathematical description provided in the previous chapter (Eq.1) [9].

In order to consider material damage, the Johnson-Cook damage model [9] is used in conjunction with the Johnson-Cook material model. Eq. 3 provides the expression for the failure plastic strain $\bar{\varepsilon}$,

$$\bar{\varepsilon} = \left(D_1 + D_2 \exp\left(D_3 \frac{\sigma}{\bar{\sigma}}\right)\right) \cdot \left(1 + D_4 \ln \frac{\dot{\bar{\varepsilon}}}{\dot{\varepsilon}_0}\right) \cdot \left[1 - D_5 \left(\frac{T - T_{room}}{T_{melt} - T_{room}}\right)^m\right]. \tag{3}$$

The fracture energy model proposed by Hillerborg [6, 3] is included in the model to determine the damage evolution. After damage initiation, the damage variable D varies by the ratio of the yield plastic deformation to the failure of plastic deformation. After damage has been initiated for $D > 0$, the new values for yield strength σ and Young's modulus $\tilde{E}$ can be calculated using $\sigma = (1 - D)\bar{\sigma}$ and $\tilde{E} = (1 - D)E$. Hence, the Johnson-Cook material model and the Johnson-Cook damage model describe linear elasticity, strain hardening and thermal softening, damage initiation, damage evolution, and eventually fracture. The used material and damage parameters are listed in Table 3.

Table 3 Johnson-Cook material parameters and damage coefficients for A2024 T351 aluminium.

A [MPa]	B [MPa]	C	n	m	$D1$	$D2$	$D3$	$D4$	$D5$
598	768	0.0137	0.2092	0.807	0.01	0.13	3.1	0	0

3.3 Contact Model and Heat Generation

The contact between tool and workpiece is described by a penalty contact algorithm. The tool is defined as master surface and the workpiece is defined as slave surface. It is to be noted that the mesh of the workpiece is fine enough to avoid penetration of the tool during contact. The Coulomb friction law is used in the contact model. It relates the frictional stress τ_f to the normal stress σ_n times a constant coefficient of friction μ, where $\mu = 0.23$ is chosen based on the previous works of [1, 21].

Since the temperature has a large impact on the accuracy of the machining simulations, a thermo-mechanical coupled problem is solved. The total heat flux $\dot{q}$ is calculated as the sum of the heat flux due to plastic strain with $\dot{q}_p = \eta\bar{\sigma}\dot{\varepsilon}^p$, and the heat flux due to friction in the tool-chip interface with $\dot{q}_f = \tau_f v_r A$. In here is η the inelastic heat fraction with η =0.9, $\bar{\sigma}$ the effective stress without damage, $\dot{\varepsilon}^p$ the plastic strain rate, τ_f the frictional stress, v_r the relative velocity and A the contact area between tool and chip. The temperature rise in the chip can be approximated as adiabatic heating according to [17] and the thermal energy generated is distributed evenly over the tool and the workpiece.

3.4 2D Micro Model of Single Scratches

This section provides information on the simulation framework of the 2D micro-cutting model, the performed benchmark simulation and sensitivity analysis. The model parameters of an orthogonal cutting process were chosen according to [25]. In an experiment by [25], a straight turning operation was performed on A2024 T351 aluminium (Young's modulus= 73 GPa, Poisson ratio= 0.3 and density=7800 kg/m^3) by a high speed steel (HSS) insert (Young's Modulus= 680 GPa, Poisson ratio= 0.33 and density= 7830 kg/m^3) with the orthogonal rake angle $\gamma = 20°$ and a clearance angle $\alpha = 5°$. The feed per revolution was set to $f = 50$ μm typically seen in grinding processes, with a constant $a_p = 1$ mm along the thickness of the workpiece and a cutting speed $v_c =$ 180 m/min. The simulation results are validated according to this experiment. In the simulation framework the element spacing d_e for the LAG/ALE models and particle spacing d_p for the SPH/PFEM models where $d_e = d_p = 0.003$ mm.

3.4.1 Benchmark Simulation

To validate the different discretization approaches LAG, ALE, SPH, and PFEM with real data, a benchmark analysis was conducted for the dry-turning operation of A2024 T351 aluminium with rake angle $\gamma = 20°$ and compared with the results from literature [25]. Fig.2 illustrates the calculated process forces and the measured forces from [25] of the reference cutting process with $\gamma = 20°$. The average measured cutting force in x-direction is 50 N in the stationary state of the experiment. The discretizations LAG, ALE and SPH predict the cutting force in good agreement with the experiment, with a minor error of 1 %. Only the PFEM approach underpredicts the cutting force with an error of 8 %. However, comparing the passive force, all the discretization approaches underpredict is with an error of 40 %. The reason for lower prediction of passive forces, in the mesh-based methods could be the settings of the built-in element deletion during material separation in the sacrificial layer. As elements are deleted that have been deformed beyond a threshold value or exceed a predefined stress value, the contact between tool and workpiece reduces and the passive force decreases. In contrast, the meshless models have a weighting function that determines the force required to release the cohesive bonds. The progression of the cohesive forces used in this work decreases in a bell-shaped manner within the area of influence, which could cause an early separation of the particles.

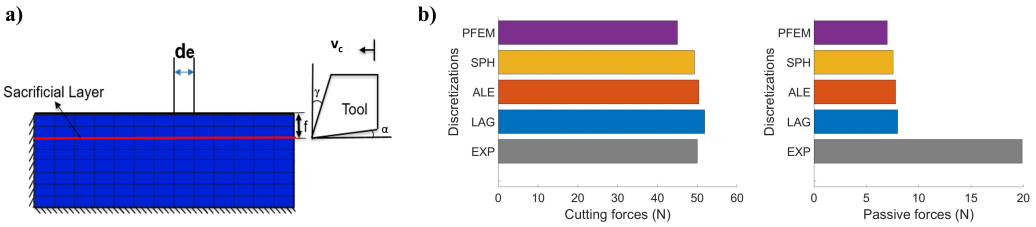


Figure 2 a) Simulation framework of the benchmark simulation and b) Comparison of the cutting/passive forces of the discretizational approaches with the experimental results from [25].

3.4.2 Sensitivity Analysis

This section presents the results of the sensitivity study when increasing the tool rake angles, towards high values between $\gamma = 0°$ to $\gamma = -45°$, typically observed in the cutting mechanisms in grinding. Comparing the process forces (cutting forces/passive forces) for the six rake angles, an increase is observed in Fig.3 with an increase in the negative rake angle γ. As the contact length at the tool-workpiece interface increases for a higher negative rake angle, this induces higher compressive stresses and plastic strains on the workpiece, which explains the increase in the process forces. The increase in the process forces is also confirmed by investigations done by [12].

4 Experimental Investigations

The grinding process involves extremely complex material removal mechanisms because of the numerous irregularly shaped and sized abrasive grits bonded onto the grinding wheel. A grinding process can be modeled as a cumulative process of the result of several single grit actions over the workpiece, thus the single grit interaction with the workpiece is of essential importance. With an understanding of individual grit-workpiece interaction mechanisms, the material removal phenomena can be extrapolated to the entire grinding wheel-workpiece interaction.

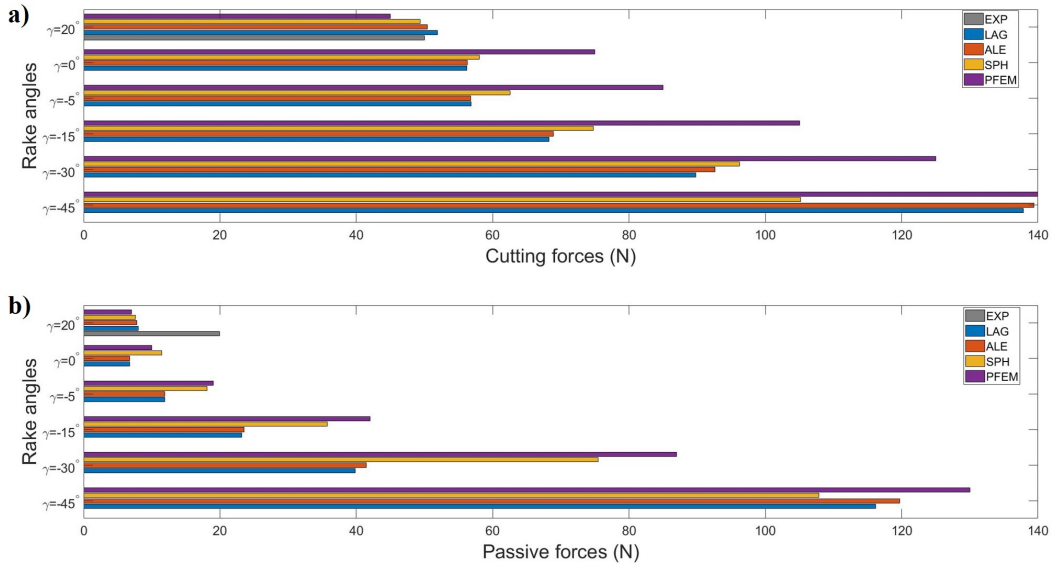


Figure 3 Comparison of the discretization approaches LAG, ALE, SPH and PFEM based on a) cutting forces and b) passive forces for the tool rake angles $\gamma = 20°$ to $\gamma = -45°$.

4.1 Experimental Setup

In this investigation, single grit grinding tests were performed on two different workpiece materials, a ductile material A2017-T4 aluminium and a brittle material soda lime glass. Scratch grooves were cut by single grit actions performed at different speeds and with gradually increasing depth of cut. A force sensor was used during the single grit grinding process to record the force exerted during scratching. The experimental setup with an exemplary force measurement is shown in Fig.4.

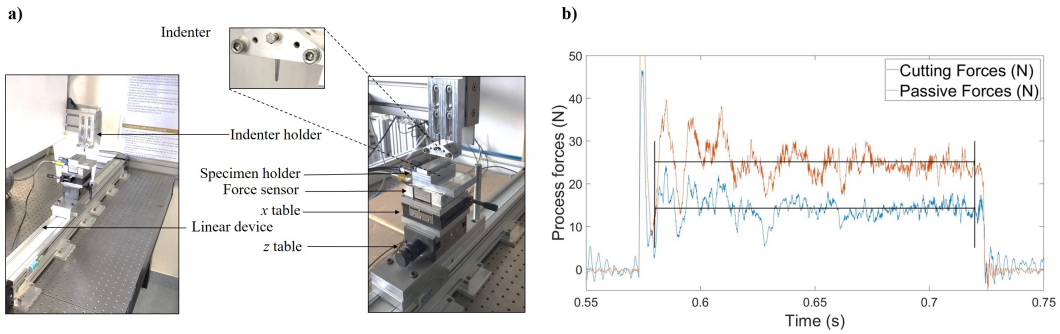


Figure 4 *a*) View of single grit scratch test setup and *b*) forces recorded with LabView software (v_c=100 mm/s on A2017 T4 workpiece).

The experimental setup consists of a linear motion device of stroke length of 850 mm. The linear axis is mounted with a z-table on the linear device, to set the depth of cut. A screw with a resolution of 1 μm drives the vertical movement. The x-table enables the motion along the horizontal direction, used to demarcate one scratch from the other. A Kistler 3-axis force sensor (Type: 9119AA1) was mounted under the workpiece to measure the forces during single grit scratching. The aluminium and glass samples are mounted on the specimen

holder with the help of vacuum, an indenter holder is fixed along a vertical column. The force measurement acquisition was performed using the LabView 2018 software package. The sampling rate was 10,000 samples/sec. The Fig.4b shows a sample measurement performed for a scratch experiment with A2017 T4 (depth of cut a_p=50 μm and linear speed v_c= 100 mm/sec). The single grain scratch tests were performed with conical indenter geometries, with cone angles 90° and 120°, as shown in Fig.5.

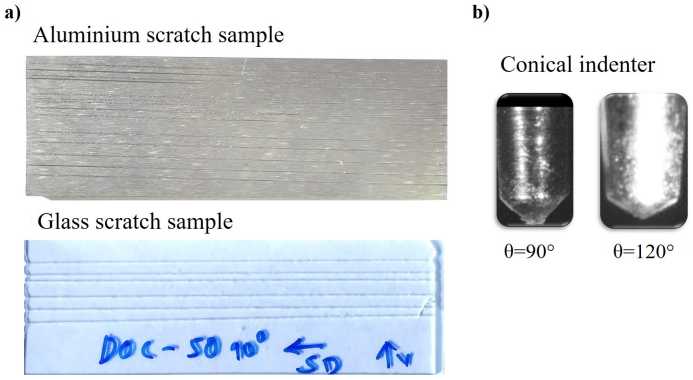


Figure 5 *a*) Single grit scratches (v_c =50 mm/s to 300 mm/s for a_p = 50 μm) on A2017 T4 and soda lime glass samples and *b*) conical indenter geometries $\theta = 90°$ and 120°.

4.2 Scratch Experiments

The scratch experiments were performed on soda lime glass samples of dimensions 75 mm × 25 mm × 1 mm. Fig.6 show process forces measured during scratch experiments performed on soda lime glass for the depth of cuts 50 μm, 70 μm and 80 μm. The scratch experiments/measurements were performed three times for all depth of cuts at various linear speeds. The statistical average of the process forces obtained, is used for the analysis. In the 3D bar plots the x-axis shows the speed of the linear device, the y-axis shows the rake angle of the indenter tool and the z-axis shows the cutting forces and passive forces in N.
According to Fig.6a, it can be seen that the cutting forces reduces with increase in speed of the linear device, for all the three depth of cuts a_p=50 μm, 70 μm and 80 μm. While comparing the cutting forces between the two conical angles (90° and 120°) for the lowest depth of cut a_p=50 μm, it is observed that the cutting forces is higher for the conical angle of 120° than for 90°. This is because the 90° conical angle appears to have the sharper tool tip, which reduces the cutting forces. However this tendency seems to invert for the higher depth of cuts a_p=70 μm and 80 μm . With the higher depth of cuts, it is observed that the 90° conical angle has undergone higher wear in comparison to the 120° conical angle. As a results the cutting action is performed by blunter cutting edges, leading to higher cutting forces measured for the indenter with the 90° conical angle.

According to Fig.6b, it can be seen that, the passive forces more or less remains constant with increase in speed of the linear device, for the depth of cuts a_p=50 μm and 70 μm. Although for a higher depth of cut of a_p=80 μm, there is a reduction in the passive forces with increase in speed of the linear device. This is observed due to formation of local damage in the scratched glass specimen. The cracks developed and propagated between the individual scratches can explain this change in the tendency. From the indenter geometry pictures in Fig.5b, we can observe that the 90° tool has higher roundness compared to the 120° tool at the tool flank. This also exerts a higher contact area on the workpiece while scratching.

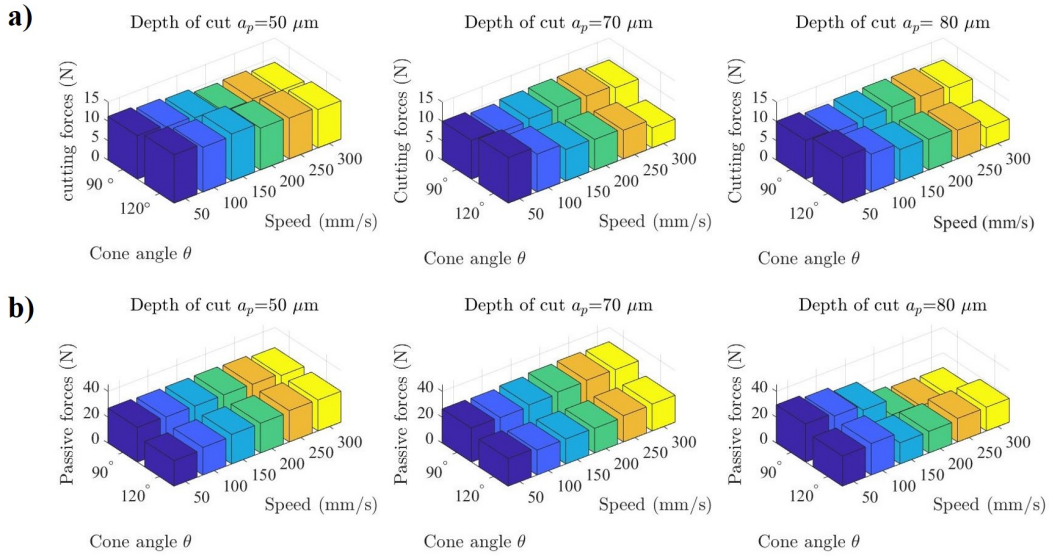


Figure 6 3D bar plots showing the measured *a*) cutting forces and *b*) passive forces, for the linear speeds v_c=50 mm/s-300 mm/s and depth of cuts a_p=50 μm, 70 μm and 80 μm.

Taking this into aspect, while comparing the passive forces between the two conical angles (90° and 120°) for depths of cut a_p=50 μm and 70 μm, the passive forces for the 120° conical angle is lower than for the 90° conical angle. Although for a higher depth of cut a_p=80 μm and for higher speeds v_c= 200-300 mm/sec the passive forces is observed to be higher for the 120° conical indenter. As the with the higher depth of cuts, it is observed that the 90° conical angle has undergone higher wear in comparison to the 120° conical angle, thereby the cutting forces increases and the passive forces decreases.
Beside the analysis of the conical angle and linear speed on the process forces of brittle soda lime glass, the effect of the speed on the process forces is additionally evaluated on aluminium 2017 T4 samples. The scratch experiments were performed on the aluminium 2017 T4 samples of dimensions 75 mm × 25 mm × 5 mm. Fig.7 shows process forces measured during scratch experiments for various speeds of the linear device v_c= 50-1000 mm/sec. According to Fig.7 and the experiments performed by [17], it can be observed that the cutting and the passive forces reduce with an increase in the speed of the linear device.

5 Virtual Grinding Wheel Modeling

Grinding wheels are one of the most important tools in the manufacturing process of high precision and high quality surfaces. This study aims to provide a realistic simulation of the manufacturing process of a grinding wheel, starting with the raw material, compression, sintering and dressing until the final grinding surface. Furthermore, it researches the effect of different levels of compression and grit sizes (F24 to F60) on the surface topography. The analysis of static grain count and bonding forces were made on the generated grinding wheel topographies. The quality of grinding wheel topographies were assessed and practical consequences were drawn for improving the manufacturing process.

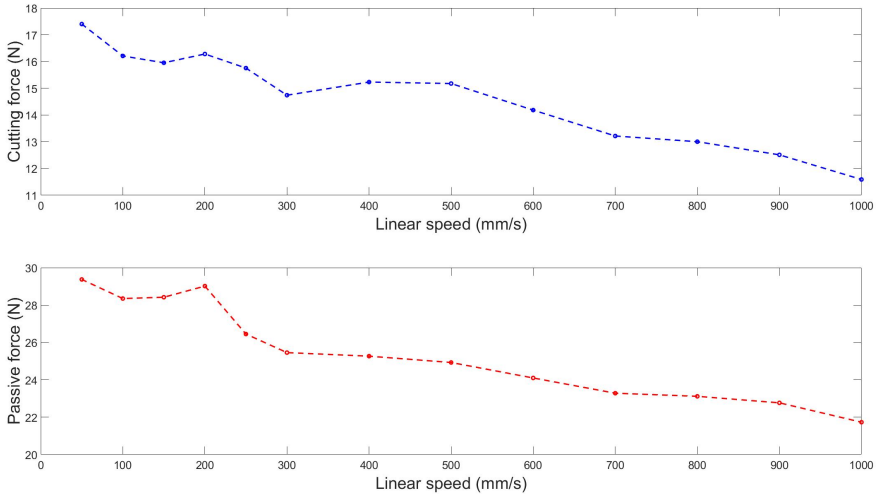


Figure 7 Process force measurements with an 120° conical angle indenter for the linear speeds v_c =50 mm/s-300 mm/s and depth of cuts a_p=50 μm on A2017 T4 specimens.

5.1 Simulation Framework

To create a realistic model, but also one that is feasible and computable without high amounts of computation power, only a cut-out of a real grinding wheel is simulated. The cut-out is displayed in Fig.8 and contains a surface-side, which is equal to the grinding surface of the grinding wheel.

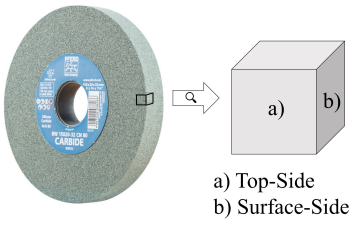


Figure 8 Grinding wheel cut-out used for simulation.

The first step in the manufacturing of a grinding wheel is weighting the raw material, i.e. the amount of bond material and abrasive particles. Afterwards, the materials are being mixed. During this process, pores are being created and their distribution is defined by the natural packing between the two components [2]. In the third step, the mixture is put into a steel form of the desired grinding wheel shape and compressed by a hydraulic press. The first three steps of the simulation were conducted with the discrete element (DEM) simulation software LIGGGHTS. After this, particles are being fired (sintered) and dressed resulting in the final grinding wheel surface. Those two steps were simulated with MATLAB, using the particle output from LIGGGHTS.

5.1.1 Particle Packing and Compression

The material mixing in the simulation was done by creating particles where the bonding material is already surrounding the grain, in comparison to the real manufacturing, where this step is happening while mixing the materials. The creation of particles in LIGGGHTS was done by a stepwise creation of new particles. The particles were randomly distributed in a certain area above the form, representing the natural distribution of the materials. The particles start falling immediately after their creation and the whole system is settled down after the creation of particles is stopped. Additional particles outside the desired shape were removed. The particles created are composite of the abrasive grain with radius R_{gi} and the correspondent amount of bond material, resulting in the total particle diameter R_{pi}, as displayed in Fig.9a .The total particle diameter R_{pi} is dependent on the distribution of bond material f_b and abrasive grain f_b, as well as the grain diameter R_{gi} [14]

$$R_{pi} = R_{gi} \cdot \left(1 + \frac{f_b}{f_g}\right)^{(1/3)} \cdot \tag{4}$$

The applied hydraulic pressure is usually between 100-5000 PSI ($0.69 - 34.5$ N/mm^2) and is applied for about 10-30 seconds [14, 8]. The simulation in this study was volume controlled, i.e. the top cylinder is moved by a certain distance per unit time towards the bottom, which is a good representation of the real manufacturing process [10]. A visual example of the compression process is displayed in Fig.9b. The maximum feasible compression is defined as the compression reached when more than 0.5 % of the particles had a radius less than their initial grain radius R_{gi}. Thus, the simulation represents the difference in material properties of the outer bond material and inner abrasive grain, as the abrasive grain is by far less compressible. Abrasive grains, e.g. silicon carbide, have a Young's modulus of 400 GPa [23], whereas the bond material, e.g. epoxy resin, only has a Young's modulus of 6.2GPa [19].

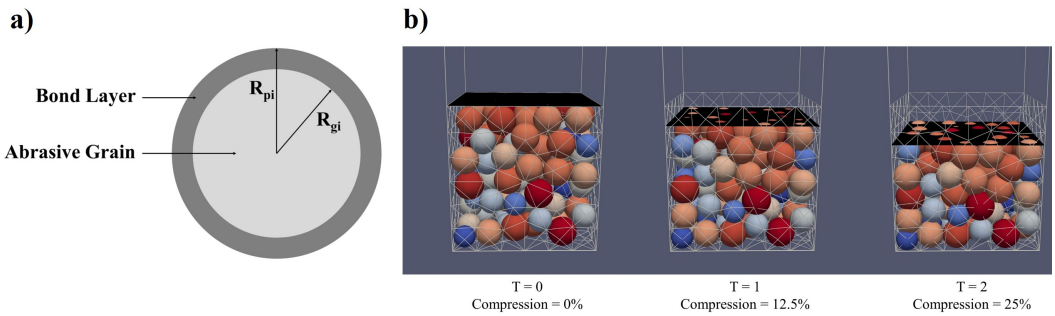


Figure 9 a) Representation of particle and bond material and b) particle packing and compression simulation.

5.1.2 Firing

The firing (sintering) of the specimen is simulated using MATLAB and uses the exported particle data from LIGGGHTS, containing the position and radius of every particle, as well as the contact surface area of the connection. The first step in the firing algorithm is the calculation of the new neck radius a, as displayed in Fig.10. The characteristic time for the grain-boundary diffusion t_g is used as a material property [27] and the sintering (firing) time for the grain-boundary diffusion [27] needs to be set. With the use of the old (initial) neck radius a_0, the formula for the new neck radius calculation is [14, 27]

$$a = a_0 \cdot \left(\frac{192t}{t_g} \right)^{(1/6)} \cdot \tag{5}$$

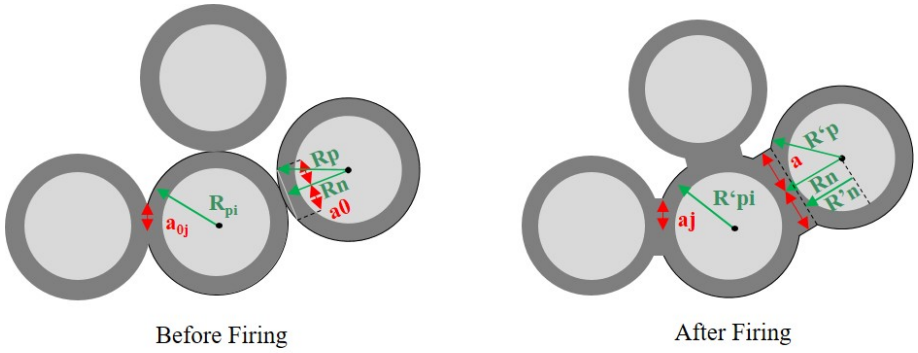


Figure 10 Change in the topology a) before and b) after the firing process with displayed characteristic geometrical values.

In Fig.10. the formation of multiple particles before and after firing is displayed. Each particle i in the set of n particles has a radius R_{pi} before firing and a radius R'_{pi} after firing. Every connection of a particle (bond j in the set of m connections has a neck radius a_{0j} before firing and neck radius a_j after firing). The calculation of the particle radius R_{ni} (see Fig.11), which connects the particle centre with the respective middle point of the neck, is calculated using Pythagoras $R_n = \sqrt{(R_p^2 - {a_0}^2)}$. The calculation of the new particle radius is based on mass conservation.

$$\left(\frac{4}{3} \cdot \pi \cdot R_{pi} \right)^3 = \left(\frac{4}{3} \cdot \pi \cdot R'_{pi} \right)^3 + \sum_{i=1}^{n} \sum_{j=1}^{m} ({a_i}^2 \pi \cdot (R_{ni} - {R'}_{ni}) - \frac{\pi}{3} {h_j}^2 \cdot (3{R'}_{pi} - h_j)) \tag{6}$$

5.1.3 Dressing

The dressing of the grinding wheel is usually conducted by a single-grit diamond [13] to reduce thickness, to correct balance or parallelism, or to create special contours (e.g. sharp edges) [14]. When the dresser hits a particle, there are three different possible situations. The first possibility is that the dresser hits the particle at its full surface, resulting in a removal of the grit, independent from dressing force or bonding force. The second option is that the dresser partly hits the particle, but the dresser force is higher than the bonding force of the particle, resulting also in a removal of the grain. The third option is that the dresser partly hits the particle, but the bonding force is higher than the force of the dresser, resulting in a grain that is being cut (broken). For the calculation of the dressing force F_t, an empirical formula was used [35]. It is based on the dressing depth a_{dp} (in mm) and the dresser travel speed f_d (in mm/min)

$$F_t = 9.631 \cdot {a_{dp}}^{-0.1992} \cdot {f_d}^{-0.1774} \tag{7}$$

The bonding force is dependent on material-specific properties for the bonding material [15]. In this study, a specific bonding force for the material of 800 N/mm^2 was used, depending on the surface area of each bonding a_j of a particle. Thus, the bonding force of a particle i is

$$F_{bi} = 800 \text{ N/mm}^2 \cdot \sum {}^{m}_{(j=1)} a_j. \tag{8}$$

5.1.4 Surface Calculation and Modelling

For the modelling of the grinding wheel topography, the software ParaView 5.7 was used. To properly display the surface, output points had to be defined in the MatLab simulation. Fig.11 displays how output points for different particles were created and how the surface was visualised using these points. In the first step, particles visible on the surface (top view) were identified. In the second step, the tip height of the particle was identified. In a third step, up to 16 additional data points were created to properly and realistically visualise the shape of the abrasive particles. Afterwards, these additionally created data points were checked for their spacing and eventually removed for smoothing the surface output and unintended spikes. In the last step the exported data points were connected and result in the topography.

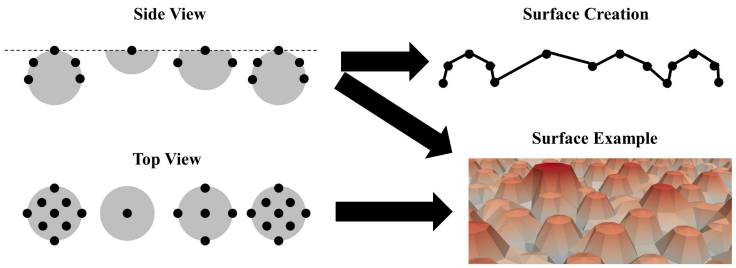


Figure 11 Visualization of the calculated surface.

All steps conducted during the simulation were visualised to enhance the understanding as well as checking for the reliability of the calculations and assumptions. Fig.12 shows an example of all the steps using the grit size 30 with a 5% compression.

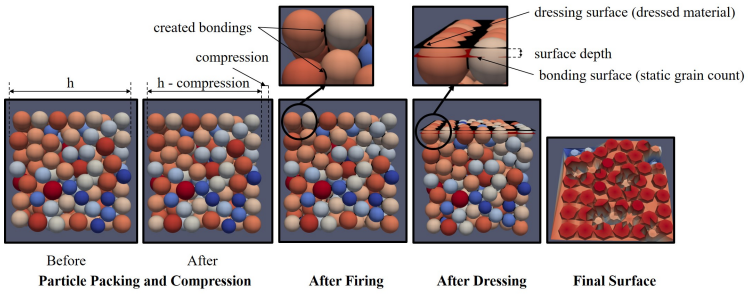


Figure 12 Visualization of the simulation steps.

5.2 Results and Discussion

The simulated grinding wheel topographies are assessed based on the output variables bonding strength and average grain count. The average bonding strength is the average of the bonding force of all the particles in the grinding wheel and their respective bondings. The static grain count is the number of grains on the wheel surface in a certain area [26].

In Fig.13 the average bonding strength is compared to different percentages of the compression strength for all grain sizes. The graph shows that an increasing particle size leads to higher average bonding strength. Thus, grit 24 had the highest maximum feasible average bonding strength with 149 N, and grit 60 the lowest, with 37 N. The characteristic slope of the bonding strength is very similar for different percentages of compression and is

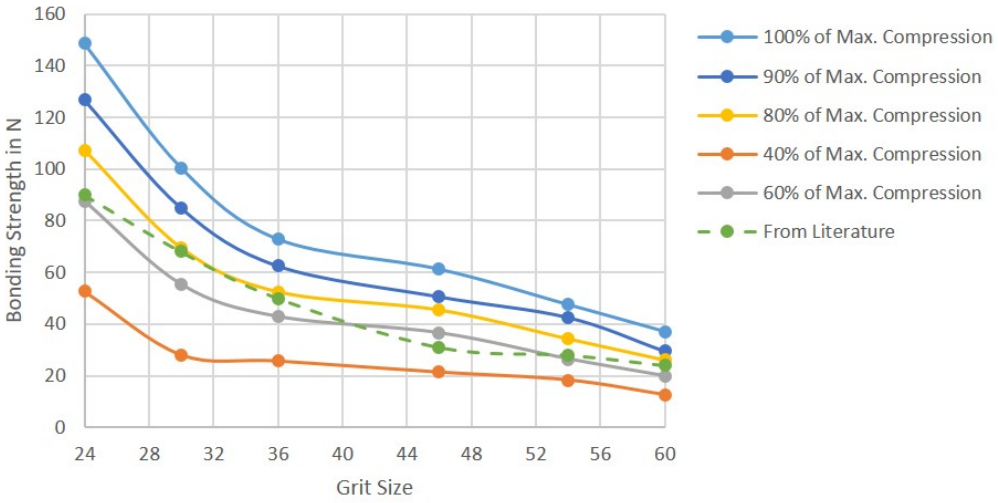


Figure 13 Average bonding strength depending on compression for different grit sizes and compared to values from [24].

very similar compared to empirical results [24]. The characteristic slope and maximum values fit well to the simulation of maximum compression, slightly varying along with different grit sizes.

The static grain count per cm^2, in comparison to the percentage of compression is displayed in Fig.14. For the maximum possible compression, the static grain count is the highest for grit 60, decreasing with grit size and having the lowest for grit 24, which is in line with empirical analysis [4], simulation [14] and theory [4]. Remarkably, all slopes for the different grit sizes show a similar characteristic. In the very beginning there is a slight, but constant increase, followed by an exponential increase, which turns into a constant value. Eventually, the slope remains constant for further compression. Thus, a value close to the maximum static grain count is reached before the maximum feasible compression.

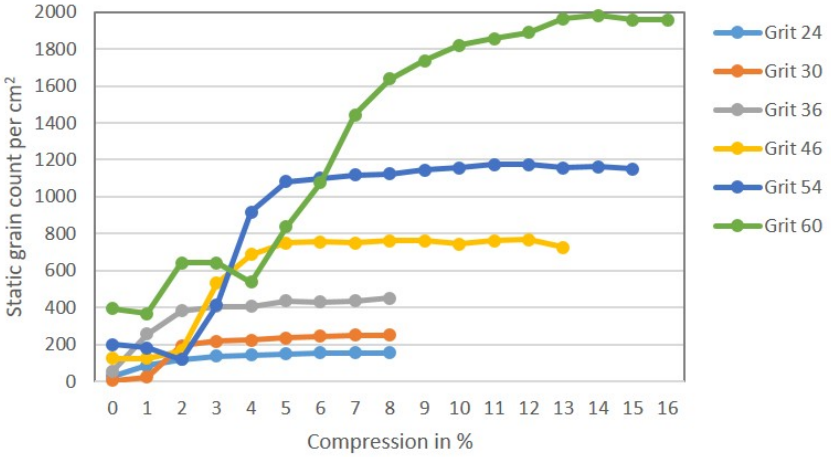


Figure 14 Static grain count dependent on different percentages of compression.

The surface calculated with Matlab was exported as a VTK file and processed using Paraview and its elevation grid filter, to provide a realistic surface of the exported data. The visualisation was enhanced by different colouring, according to height, bonding force, protrusion height and alternative protrusion height. The standard surface representation is coloured based on the lowest surface point in the respective area along with all compression percentages. It is visible that the surface is rather unstructured for low percentages of compression and a more even and structured surface is created with higher compression (see Fig.15). A very even and uniform surface is usually reached above 50 % of the maximum compression. It is also visible that with increasing compression, random surface caverns start disappearing, reducing the pore volume of the surface, as visible in Fig.15 Low percentages of compression might contain single grains that protrude from the surface which might leave unintended scratches while grinding. The surface topography appears to be similar to previous surface simulations with [14].

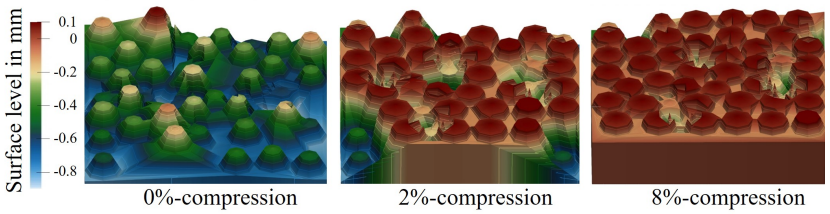


Figure 15 Exemplary surface topography of grit 30 for 0 %, 2% amd 8 % compression. Colormap indicates height above/below surface level in mm.

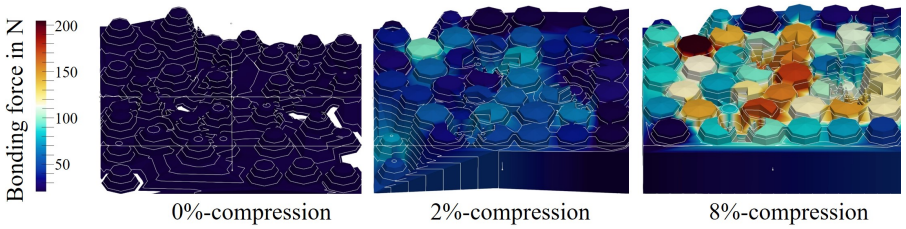


Figure 16 Bonding force of grit 30 for 0%, 2% and 8% compression. Colormap indicates the bonding forces in N.

To analyse the bonding strength of the grinding wheel, the surface was coloured to the respective bonding strength of a particle and the surrounding area, see Fig 16. For all grit sizes the characteristic development of the bonding force and the distribution is similar, just as the analysis of the bonding strength has shown. While the bonding force is very low at low compression, it increases with increasing compression. Although the distribution is evenly in the beginning, at higher compressions the differences in bonding force of the particles are visible. Though occasional particles stand out with a higher bonding force, but particles with low and higher bonding forces are equally distributed across the surface area.

6 Overview and Future work

To give a brief overview of the paper, four individual research studies were performed, and the most important findings are as follows:

- The first study was performed to parameterize the JC-model with respect to a micro indentation experiments. EBSD and CPFEM simulations were performed to understand

if the micro-structure of the workpiece material has an influence in material removal mechanism in a micro-indentation process. The fitting to the experimental data is only a little better for CPFEM model when compared to the JC-model. In the future studies, an extension to this comparative study will be done for a single grain scratch process, to understand the capabilities of both models to describe complex material removal mechanisms.

- In the second study, the influence of higher tool rake angles on the process forces in a micro-cutting process was analyzed with the help of a 2D orthogonal cutting model. Four different discretizational approaches (LAG, ALE, SPH and PFEM) were employed to model the workpiece A2024 T351 and to understand which approach is most suitable for metal cutting simulations. As a conclusion to this study, all the discretizational approaches were benchmarked with micro-cutting experiments found in literature [25]. The simulation models successfully predict the cutting forces with a maximum error of 8 % and the passive forces with an error of 40 %. From the sensitivity analysis performed with higher tool rake angles, the tendency is observed that the process forces increase with an increase in the negative tool rake angle from $\gamma = 20°$ to $\gamma = -45°$. This tendancy is also confirmed by the investigation performed in [12]. The model needs to be extended to a 3D simulation to also simulate material removal mechanisms such as plowing and rubbing, which is typically observed in a grinding/scratching process.
- In the third study, scratch experiments was performed to analyse the material removal process for a ductile material aluminium A2017 T4 and a brittle material soda lime glass. The experiments help to analyse the effect of change in the process forces (cutting forces, passive forces) under the influence of change in linear speed and changes in the depth of cuts a_p. Two conical indenter geometries ($\theta = 90°$ and $\theta = 120°$) were used to understand the effect of the change in the tool geometry. During the single grit scratching of glass, it is observed that the cutting forces reduce and the passive forces remains constant with increasing linear speeds. The cutting forces measured for the indenter geometries $\theta = 120°$ is higher in comparison to $\theta = 90°$, whereas the passive forces show an opposite tendancy. While performing the scratch experiments with aluminium, it was observed that the process forces reduce with the increase in the linear speed, which is also confirmed by literature [17].
- Finally for the fourth study, a realistic simulation of the manufacturing process of a grinding wheel was carried out, starting with the raw material, compression, sintering and dressing until the final grinding surface was developed. The bonding strengths and static grain count were analyzed and the tendencies were observed that an increasing percentage of compression was found to increase the bonding strength of particles. Furthermore, the larger the size of the abrasive particles, the higher is the bonding strength. The static grain count was found to increase with increasing compression and with decreasing particle size.

In conclusion, the first three research studies focuses on the microscopic scale of single grit-workpiece interactions. These studies help in the understanding of the single grain scratch process and thereby contributes to the final goal of developing a physical force model for single grit scratching. In future, these studies will be extended to enhance the capabilities of the physical force model for the single grit approach. The final study focuses on the macroscopic scale of grinding-workpiece interactions. The development of a virtual wheel topography is the first step that aids for upscalling the physical force model from the single grit scratch process to a complete grinding process.

References

1 P.J. Arrazola. *Modelisation numerique de la coupe: etude de sensibilite des parametres d'entree et identification du frottement entre outil-copeau.* PhD thesis, L'École Centrale de Nantes, 2003.

2 C.P. Bhateja and R.P. Lindsay. The importance of abrasive grinding wheel hardness control for the productivity of production grinding operations. *CIRP Annals*, 30(1):247–249, 1981.

3 B. Borsos, A. Csörgő, A. Hidas, B. Kotnyek, and G. Stépán. Two-Dimensional Finite Element Analysis of Turning Processes. *Periodica Polytechnica Mechanical Engineering*, 61(1):44–54, 2017.

4 X. Chen and W.B. Rowe. Analysis and simulation of the grinding process. part i: generation of the grinding wheel surface. *International Journal of Machine Tools and Manufacture*, 36(8):871–882, 1996.

5 D.A. Doman, R. Bauer, and A. Warkentin. Experimentally validated finite element model of the rubbing and ploughing phases in scratch tests. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 223(12):1519–1527, 2009.

6 A. Hillerborg, M. Modéer, and P. E. Petersson. Analysis of crack formation and crack growth in concrete by means of fracture mechanics and finite elements. *Cement and Concrete Research*, 6(6):773–781, 1976.

7 S. R. Idelsohn, E. Onate, and F. Del Pin. The particle finite element method: A powerful tool to solve incompressible flows with free-surfaces and breaking waves. *International Journal for Numerical Methods in Engineering*, 61(7):964–989, 2004.

8 M.J. Jackson. *A study of vitreous-bonded abrasive materials.* PhD thesis, Liverpool John Moores University, 1995.

9 G.R. Johnson and W.H. Cook. A constitutive model and data for metals subjected to large strains, high strain rates and high temperatures. In *Proceedings of the 7th International Symposium on Ballistics*, pages 541–547, 1983.

10 B. Kirsch, C. Effgen, M. Büchel, and J.C. Aurich. Comparison of the embodied energy of a grinding wheel and an end mill. *Procedia CIRP*, 15:74–79, 2014.

11 M.W Klein, B. Blinn, M. Smaga, and T. Beck. High cycle fatigue behavior of high-mn twip steel with different surface morphologies. *International Journal of Fatigue*, 134:105499, 2020.

12 S. Kosaraju, V. Anne, and V. Ghanta. Effect of Rake Angle and Feed Rate on Cutting Forces in an Orthogonal Turning Process. *International Conference on Trends in Mechanical and Industrial Engineering*, 61(May):150–154, 2011.

13 M. Li, W. Ding, B. Li, and J. Xu. Morphological evolution and grinding performance of vitrified bonded microcrystal alumina abrasive wheel dressed with a single-grit diamond. *Ceramics International*, 45(16):19669–19678, 2019.

14 X. Li. *Modeling and simulation of grinding processes based on a virtual wheel model and microscopic interaction analysis.* PhD thesis, Worcester Polytechnic Institute, 2010.

15 M. Hrairi M. Djemana. Modelling and Simulation of Impedance-Based Damaged Monitoring of Structures. *International Journal of Modelling and Simulation*, 15:395–408, 2016.

16 L. Olovsson, L. Nilsson, and K. Simonsson. ALE formulation for the solution of two-dimensional metal cutting problems. *Computers and Structures*, 72(4):497–507, 1999.

17 T. Opoz. *Investigation of Material Removal Mechanism in Grinding: A Single Grit Approach.* PhD thesis, University of Huddersfield, 2012.

18 D.T. Pierce, K. Nowag, A. Montagne, J.A. Jiménez, J.E. Wittig, and R. Ghisleni. Single crystal elastic constants of high-manganese transformation-and twinning-induced plasticity steels determined by a new method utilizing nanoindentation. *Materials Science and Engineering: A*, 578:134–139, 2013.

19 Manika Prasad, Malgorzata Kopycinska, Ute Rabe, and Walter Arnold. Measurement of young's modulus of clay minerals using atomic force acoustic microscopy. *Geophysical Research Letters*, 29(8):13–1–13–4, 2002.

20 J.M. Rodríguez, J.M. Carbonell, and P. Jonsén. Numerical Methods for the Modelling of Chip Formation. *Archives of Computational Methods in Engineering*, pages 1–48, 2018.

21 J.M. Rodríguez, J.M. Carbonell, J C. Cante, J. Oliver, and P. Jonsén. Generation of segmental chips in metal cutting modeled with the PFEM. *Computational Mechanics*, 61(6):639–655, 2018.

22 F. Roters, P. Eisenlohr, L. Hantcherli, D.D. Tjahjanto, T.R. Bieler, and D. Raabe. Overview of constitutive laws, kinematics, homogenization and multiscale methods in crystal plasticity finite-element modeling: Theory, experiments, applications. *Acta Materialia*, 58(4):1152–1211, 2010.

23 James F Shackelford. *Introduction to materials science for engineers*. Pearson Upper Saddle River, 2016.

24 M. C. Shaw. Principles of abrasive processing. In *Proceedings of the 32nd International MATADOR Conference*, 1996.

25 S. Subbiah. *Some investigations of scaling effects in micro-cutting*. PhD thesis, Georgia Institute of Technology, 2006.

26 S. Xu, D. Ruan, J.H. Beynon, and Y. Rong. Dynamic tensile behaviour of twip steel under intermediate strain rate loading. *Materials Science and Engineering: A*, 573:132–140, 2013.

27 Y.Y. Zhu, S.H. Liang, Z.J. Zhan, P. Xiao, and Z.K. Fan. Simulation of the change of sintering neck between two grains in two dimensions. *Acta Metallurgica Sinica (English Letters)*, 19(6):397–404, 2006.

Molecular Simulation Study on the Influence of the Scratching Velocity on Nanoscopic Contact Processes

Sebastian Schmitt
Laboratory of Engineering Thermodynamics (LTD), TU Kaiserslautern, Germany

Simon Stephan[1]
Laboratory of Engineering Thermodynamics (LTD), TU Kaiserslautern, Germany

Benjamin Kirsch
Institute for Manufacturing Technology and Production Systems (FBK), TU Kaiserslautern, Germany

Jan C. Aurich
Institute for Manufacturing Technology and Production Systems (FBK), TU Kaiserslautern, Germany

Eberhard Kerscher
Materials Testing (AWP), TU Kaiserslautern, Germany

Herbert M. Urbassek
Physics Department and Research Center OPTIMAS, TU Kaiserslautern, Germany

Hans Hasse
Laboratory of Engineering Thermodynamics (LTD), TU Kaiserslautern, Germany

Abstract

The influence of the scratching velocity on mechanical and thermal properties of a nanoscopic contact process was studied by molecular dynamics simulations. Simulations with different scratching velocities were conducted in dry and lubricated systems. The contact process consisted of a lateral scratching of a spherical indenter on a planar substrate. All molecular interactions were described by the Lennard-Jones truncated and shifted potential. The forces on the indenter, the coefficient of friction and the work done by the indenter as well as the power applied on the indenter were sampled. Furthermore, an analysis of thermal properties was conducted: The change of the energy of the substrate, the indenter and the fluid was evaluated and the local temperature field was determined. The forces, the coefficient of friction and the work done by the indenter show practically no influence of the scratching velocity. The work done by the indenter was found to be the same for all velocities. As a consequence, the power supplied to the system depends linearly on the scratching velocity, which affects the temperature of the contact zone. As expected, the presence of a lubricant reduces the temperature of the substrate in the vicinity of the contact.

2012 ACM Subject Classification Applied computing → Physical sciences and engineering

Keywords and phrases Nanotribology, Friction, Scratching, Lubrication, Lennard-Jones Potential

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.17

Funding The authors gratefully acknowledge financial support by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057.

Acknowledgements The present work was conducted under the auspices of the Boltzmann-Zuse Society of Computational Molecular Engineering (BZS). The simulations were carried out on the Regional University Computing Center Kaiserslautern (RHRK) under the grant TUK-MTD, the High Performance Computing Center Stuttgart (HLRS) under the grant MMHBF2 as well as the Leibniz Supercomputing Centre (LRZ) under the grant (AMSEL)ˆ2 (pn56mo).

[1] Corresponding author

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 17; pp. 17:1–17:16

OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Contact processes play an important role in different fields of technology, e.g. during machining processes between the cutting tool and the workpiece and in bearings between roughness asperities of two macroscopic surfaces. Because of the small dimensions of the actual contact zone, the processes therein can barely be investigated experimentally and, therefore, molecular dynamics (MD) simulations are usually used to investigate nanoscopic contacts [42]. In molecular dynamics simulations, Newton's equations of motion are solved for an atomistic many-particle system. The interactions of the particles in molecular dynamics simulations are often described by classical force fields. Molecular dynamics simulations are known to have good predictive capabilities and are widely used to simulate material properties [23, 39, 41] as well as nanoscale processes [15]. In molecular simulations of contact processes, the macroscopic contact is usually reduced to a single surface asperity in contact with a flat surface [13]. The actual contact process is usually modeled by a sequential movement of the indenter: (1) The indenter is brought into contact with the substrate during the indentation phase; (2) the indenter carries out a lateral scratching through the substrate surface during the scratching phase. The present work studies the influence of the indenter velocity during the scratching.

There are many studies focusing on different aspects of dry nanoscratching. They address the influence of the indenter geometry [2], different substrate materials [1, 49], the surface orientation [16] and dislocations [18]. There are few studies available in the literature addressing the comparison of lubricated and dry nanoscratching processes [27,28,34,35,38,40]; most studies on nanoscopic contact processes do not consider the presence of a fluid as lubricant and cooling agent [45]. The influence of the lubricant on both mechanical and thermal properties like the coefficient of friction or the temperature field were studied in the present work at different scratching velocities.

The influence of the scratching velocity in nanoscopic scratching processes was subject of several studies [19, 29, 31, 32, 47, 48, 50], however, mostly for dry contacts. Only Ren et al. [34] investigated the influence of the scratching velocity on mechanical properties (the forces on the indenter and the coefficient of friction) in a system with a thin water film as lubricant. Overall, scratching velocities in the range of 1 – 400 m/s were applied [34, 48]. The influence of the scratching velocity on different mechanical properties were investigated, whereas little is known on the influence on thermal properties. Only the influence of the scratching velocity on the temperature of the substrate has been studied [29, 31]. Most authors report a decrease of the coefficient of friction with increasing scratching velocity in the dry case [32, 50]. As expected, the temperature in the contact zone increases with increasing scratching velocity [19, 48]. For lubricated systems, Ren et al. [34] found that the coefficient of friction does not depend on the scratching velocity. Besides the influence of the velocity in a scratching process, the influence of the velocity on a process of indentation has been studied more thoroughly, e.g. Refs. [3, 17, 36].

Most molecular simulation studies on contact processes investigate mechanical properties focusing on the impact on the substrate, e.g. elastic and plastic deformation, the dislocation behavior, the friction forces and the coefficient of friction [18, 49]. Thermal properties such as the temperature field in the contact zone or the energy balance of the process have been investigated only by few authors [35, 38]. The energy balance of the process is of particular interest in the lubricated case to evaluate the cooling capability of the fluid, which is of fundamental importance in tribological and machining applications [7, 24, 30].

In the present study, molecular dynamics simulations of a generalized model system were used to study the influence of the scratching velocity on both mechanical and thermal properties. Both dry and lubricated cases were considered. By systematically investigating

the influence of the scratching velocity on both dry and lubricated contact processes, this study stands out from previous studies in the literature. The lateral scratching movement of a spherical indenter on a planar surface was analysed in detail. All occurring interactions were modeled with the Lennard-Jones truncated and shifted (LJTS) potential. The LJTS potential is frequently used in the literature [12, 21, 26] as it is computationally particularly cheap and can describe the properties of small, spherical molecules well [46]. The studied scratching velocities are in the range of 20 m/s – 100 m/s, which are typical conditions in grinding processes in manufacturing applications [25]. Within this range, five different velocities were chosen and applied in both the dry and the lubricated case.

2 Methods

2.1 Molecular Model

The simulation scenario includes three different substances: The solid substrate, the spherical solid indenter and the fluid (cf. Fig. 1). The latter is only present in the lubricated cases.

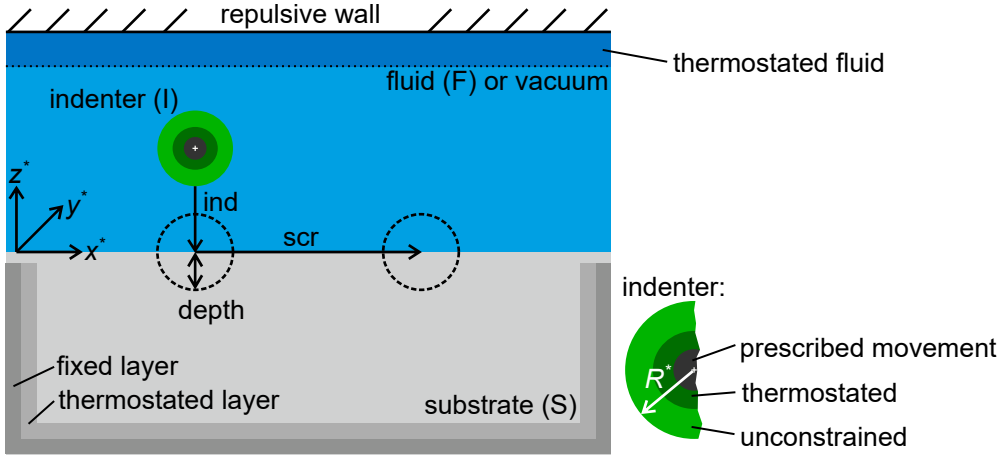


Figure 1 Simulation scenario (not true to scale) with substrate (grey), indenter (green) and vacuum (dry cases, blue) or fluid (lubricated cases, blue). Simulation procedure: First indentation (ind) of indenter, then scratching (scr).

All occurring intermolecular interactions are modeled with the LJTS potential

$$U_{\mathrm{LJTS}}(r) = \begin{cases} U_{\mathrm{LJ}}(r) - U_{\mathrm{LJ}}(r_{\mathrm{c}}) & r \leq r_{\mathrm{c}} \\ 0 & r > r_{\mathrm{c}} \end{cases} \quad \text{with} \quad U_{\mathrm{LJ}}(r) = 4\,\varepsilon \left[(\sigma/r)^{12} - (\sigma/r)^{6} \right], \tag{1}$$

where σ is the size parameter, ε the energy parameter and r the distance between two particles. The full Lennard-Jones potential is denoted by $U_{\mathrm{LJ}}(r)$. The cut-off radius is $r_{\mathrm{c}} = 2.5\,\sigma$ for all site-site interactions except the solid-solid interaction between the substrate and the indenter particles. For the solid-solid interaction the cut-off radius is chosen as $r_{\mathrm{c}} = 2^{1/6}\,\sigma$ such that the indenter-substrate interactions are purely repulsive (the potential is truncated at the potential minimum). All occurring physical quantities in this work are reduced using the parameters of the potential of the fluid particles σ_{F}, ε_{F} and the mass of a fluid particle m_{F} [4] using the relations given in Table 1. The size parameter σ of the potentials of all three substances as well as the masses m of all particles were equal.

Table 1 Definition of physical quantities in reduced units. The asterisk (*) indicates a reduced quantity. k_B is the Boltzmann constant.

Length	$L^* =$	L/σ_F
Temperature	$T^* =$	$T/(\varepsilon_\mathrm{F}/k_\mathrm{B})$
Force	$F^* =$	$F/(\varepsilon_\mathrm{F}/\sigma_\mathrm{F})$
Time	$t^* =$	$t/(\sigma_\mathrm{F}\sqrt{m_\mathrm{F}/\varepsilon_\mathrm{F}})$
Velocity	$v^* =$	$v/\sqrt{\varepsilon_\mathrm{F}/m_\mathrm{F}}$
Energy	$E^* =$	E/ε_F
Power	$P^* =$	$P/\sigma_\mathrm{F}\sqrt{m_\mathrm{F}/\varepsilon_\mathrm{F}^3}$

Table 2 shows the energy parameters for the interactions between particles of the substrate, the indenter and the fluid. Both interactions with particles from the same substance (diagonal elements) and with particles from another substance (off-diagonal elements) are included in Table 2. The energy parameter for the fluid ε_F was chosen such that it models the thermodynamic properties of methane [46]. The energy parameter of the substrate particles ε_S was adopted from Ref. [20] such as to represent iron. The energy parameter for the indenter ε_I was chosen sufficiently large such that no plastic deformations or dislocations occur in the indenter and that there is no wear of the indenter in the process (this was tested in preliminary simulations). The chosen solid-fluid interaction energy between the fluid and the substrate particles $\varepsilon_\mathrm{S\text{-}F}$ as well as between the fluid and the indenter particles $\varepsilon_\mathrm{I\text{-}F}$ was chosen to be in the domain of total wetting [5]. All simulations were carried out using LAMMPS [33].

Table 2 Pairwise interaction energy parameters used in the simulations. The size parameter σ of all potentials and the molar mass M of all particles were $\sigma = 3.7241$ Å [46] and $M = 16.043$ g/mol, respectively.

	Substrate	Indenter	Fluid
Substrate	$\varepsilon^*_\mathrm{S} = 34.89$	$\varepsilon^*_\mathrm{S\text{-}I} = 300$	$\varepsilon^*_\mathrm{S\text{-}F} = 1.5$
Indenter		$\varepsilon^*_\mathrm{I} = 200$	$\varepsilon^*_\mathrm{I\text{-}F} = 1.5$
Fluid			$\varepsilon_\mathrm{F} = 175.06$ K/k$_\mathrm{B}$

2.2 Simulation Setup

The dimension of the simulation box in x^*, y^* and z^*-direction was $L^*_\mathrm{x} = 241.30$, $L^*_\mathrm{y} = 120.64$ and $L^*_\mathrm{z} = 194.81$. In x^* and y^*-direction, periodic boundary conditions were applied. At the top, the simulation box was enclosed by a soft repulsive wall with a harmonic potential. The substrate block comprised the whole x^*-y^* plane and had a height of $\Delta z^* = 116.56$. The principles of the simulation setup were adopted from Ref. [40]. The simulation procedure was as follows: In a first step, the initial density of the fluid ($\rho^* = 0.804$, which is a liquid state at the given temperature) was adjusted (only in the lubricated cases) by moving the upper repulsive wall in negative z^*-direction. Subsequently, the entire simulation box was equilibrated at the initial temperature $T^* = 0.8$. The equilibration was carried out for 5×10^5 time steps. The time step was $\Delta t^* = 0.0008$ in all simulations. After the equilibration, the indentation started by moving the indenter in negative z^*-direction until the depth of penetration $z^* = -6$ was reached (cf. Fig. 1). Two different simulation setups were

considered in the present work: Dry cases and lubricated cases. To study the influence of the scratching velocity, two base simulations were equilibrated and indented: One dry and one lubricated. These two equilibrated setups were used in several scratching simulations. Hence, the initial conditions for each of the dry and lubricated cases, respectively, were identical at the beginning of the scratching process. The indenter scratched in lateral, i.e. positive x^*-direction, for a total length of $L^*_{\mathrm{max}} = 80.56$ in all simulations. Five different scratching velocities $v^*_{\mathrm{scr}} = 0.066, 0.133, 0.204, 0.266$ and 0.332 were studied – each for the dry and the lubricated cases. These scratching velocities correspond to 20, 40, 60, 80 and 100 m/s (using the potential parameters of Table 2).

The origin of the Cartesian coordinate system was defined such that $z^* = 0$ was the height of the initial undeformed surface of the substrate and $x^* = 0$ and $y^* = 0$ were located in the corner of the simulation box. The substrate consisted of 3.65×10^6 particles arranged on a single crystal face cubic centered (fcc) lattice, which is the equilibrium lattice at the given state [37, 43]. The surface orientation was (100). The scratching was conducted in [1 1 0] surface direction. To fix the substrate block in place, layers with fixed atom positions were employed at the simulation box boundaries. To model the heating of the surrounding material, thermostated layers were located between the actual substrate and the fixed layers (cf. Fig. 1).

The spherical indenter had a radius of $R^* = 12.08$ and, similar to the substrate, consisted of three parts. For particles in the core ($R^* \leq 2.69$), the motion was prescribed to realize the relative motion of the indenter with respect to the substrate. Around the core, there was a thermostated shell with a thickness of $\Delta R^* = 2.69$. The particles of the outer shell ($R^* > 5.38$) were unconstrained and moved freely following Newton's equations of motion. The thickness of the thermostated layer was chosen such that the particles of the outer shell did not interact with the rigid core. The indenter consisted of 7935 particles which were, as the substrate, arranged on an fcc lattice.

In the lubricated cases, the indenter was fully immersed in the fluid, which consisted of 1.78×10^6 particles. A fluid region at the top of the simulation box below the soft repulsive wall ($62.14 < z^* < 78.25$) was thermostated. The thermostating in all three thermostated zones (substrate, indenter and fluid) was carried out using the Berendsen thermostat [6] to constrain their temperature constant at $T^* = 0.8$. The thermostated layers removed the dissipated thermal energy from the simulation box.

2.3 Definition of Observables

To examine the influence of the scratching velocity on the contact process and to compare the results from the dry and the lubricated cases various mechanical and thermal properties were used. They are defined in the following. The total tangential and normal forces on the indenter F^*_{t} and F^*_{n} were computed as the sum of the forces in x^* and z^*-direction acting on all indenter particles, respectively. From the forces on the indenter, the coefficient of friction was calculated from $\mu = -F^*_{\mathrm{t}}/F^*_{\mathrm{n}}$. The work done by the indenter was calculated from $W^*_{\mathrm{I}} = \int -F^*_{\mathrm{t}} \mathrm{d}x^*$. The power supplied to the system was calculated from $P^*_{\mathrm{I}} = \mathrm{d}W^*_{\mathrm{I}}/\mathrm{d}t^*$.

Furthermore, the change of the energy ΔE^*_j in the three parts of the simulation box was calculated as: $\Delta E^*_j = \Delta U^*_j + \Delta E^*_{\mathrm{thermo},j}$ for j = S, I, F (substrate, indenter and fluid). The change of the energy ΔE^*_j is composed of the change of the internal energy ΔU^*_j (sum of kinetic and potential energy) of the respective part and the energy removed by the respective thermostat $\Delta E^*_{\mathrm{thermo},j}$. The change of the internal energy ΔU^*_j includes the change of the kinetic energy of the respective parts as well as the energy of the elastic and the plastic deformation (only for the solids), which is part of the potential energy. For simplicity, all these

contributions are summarized in the single observable ΔE_j^* although this reflects different phenomena of the process such as the dissipation of the work done on the system, elastic and plastic deformation of the solids and the heat flow to the surroundings as represented by the action of the thermostats in the thermostated zones. The corresponding change of the energy per time $\mathrm{d}E_j^*/\mathrm{d}t^*$ is defined as the derivative of the change of the energy with respect to time.

To characterize the thermodynamic behavior of the system, the temperature in the contact zone T^*_{contact} was evaluated as the mean temperature of the substrate particles in a volume V^*_{contact} in front of the indenter (cf. Fig. 2), whereas the directed velocity from the particles is subtracted from the total average kinetic energy for the computation of the local temperature by the thermal motion, cf. Ref. [38] for details. The contact volume V^*_{contact} has the dimensions $2\,R^* \times 2\,R^* \times 3\,R^*$, where R^* is the radius of the indenter. Furthermore, the temperature distribution in the x^*-z^* plane was sampled during the simulations: A stripe at $y^* = L^*_{\mathrm{y}}/2$ with an extend of $\Delta y^* = 2.15$ was segmented into cuboid bins (with $\Delta x^* = \Delta z^* = 1.34$), in which the temperature was averaged over 5000 time steps in every bin (cf. Fig. 2). The gap between the indenter and the substrate was defined as the space between the two bodies below the undeformed surface level and in front of the center of the indenter. This lubrication gap definition is used to count the fluid particles trapped between the two solid bodies, cf. Fig. 2.

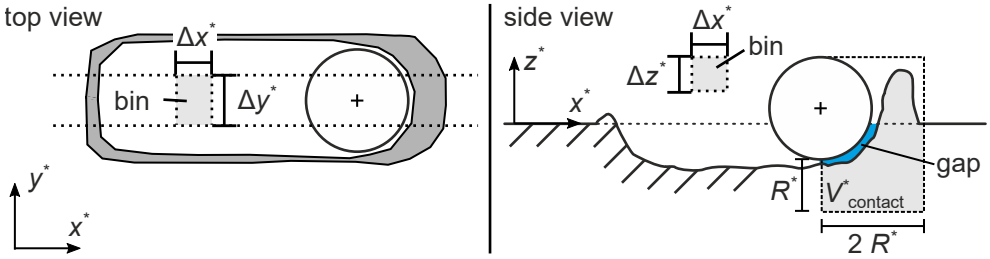


Figure 2 Scheme for illustrating the observables and bins to spatially evaluate the temperature. Left (top view): Definition of the bins in the x^*-y^* plane with $\Delta x^* = 1.34$ and $\Delta y^* = 2.15$. Right (side view): Definition of the gap between the indenter and the substrate (blue), the volume V^*_{contact} (light grey) and the bins in the x^*-z^* plane with $\Delta x^* = \Delta z^* = 1.34$.

3 Results and Discussion

3.1 Mechanical Properties

The results for the tangential and normal forces on the indenter F_{t}^* and F_{n}^* as well as the coefficient of friction μ as a function of the scratching length L^*_{scr} are shown in Fig. 3. At the beginning of the scratching ($L^*_{\mathrm{scr}} < 5$), the tangential force F_{t}^* in the dry and the lubricated cases shows a steep decrease. During the ongoing scratching process, the tangential force fluctuates in the range $-4 < F_{\mathrm{t}}^*/10^3 < -2$ in the dry and the lubricated cases. As expected, the value of the tangential force is negative such that it acts as a force against the direction of motion. No significant differences between the results for the dry and lubricated cases are observed. The scratching velocity has no significant influence on the forces of the contact process in the studied range of scratching velocities.

The normal force F_{n}^* in the dry cases is approximately 6.5 at the beginning of the scratching and decreases within $L^*_{\mathrm{scr}} < 10$ to values in the range $2 < F_{\mathrm{n}}^*/10^3 < 5$. The normal force in the lubricated cases has a similar behavior. The values of F_{n}^* after the indentation ($L^*_{\mathrm{scr}} = 0$) are higher in the lubricated cases due to fluid particles confined

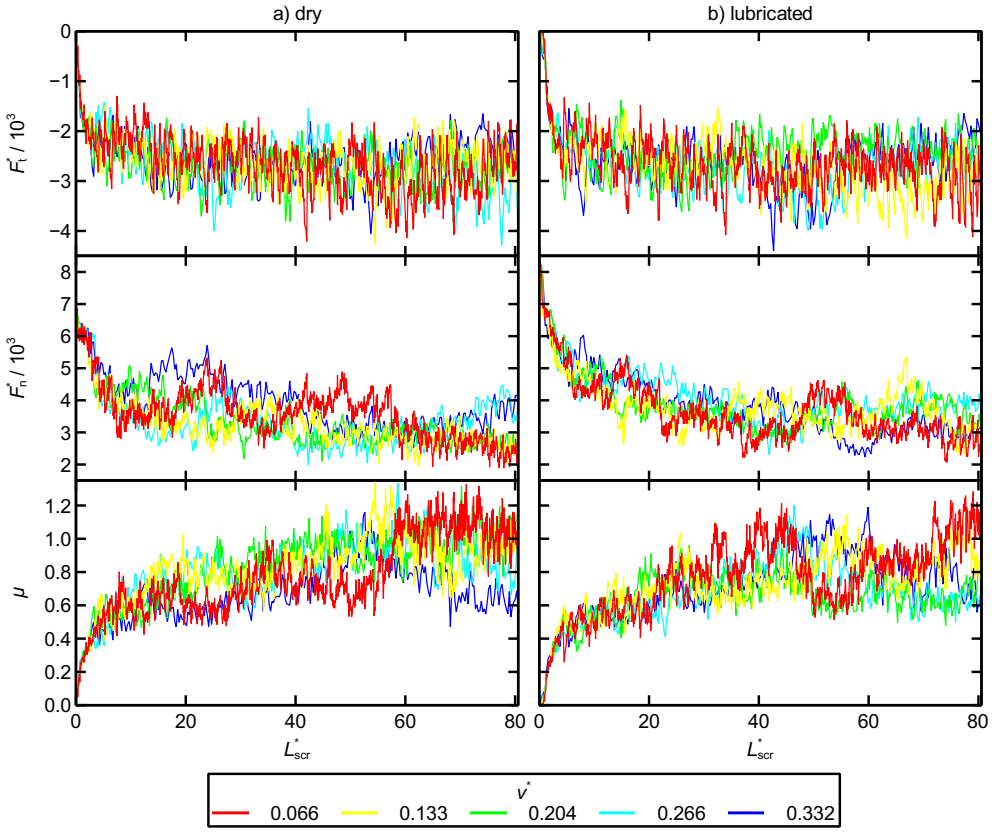


Figure 3 Tangential force F_t^* (top), normal force F_n^* (middle) on the indenter and coefficient of friction μ (bottom). Results of simulations for dry cases (left) and lubricated cases (right) from all five considered scratching velocities as function of the scratching length L_{scr}^*.

between the indenter and the substrate, cf. Ref. [40] for a detailed discussion. Additionally, the decrease of the normal force F_n^* is less steep in the lubricated cases and only reaches a constant level for $L_{scr}^* > 20$, where the values are in the range of $2 < F_n^*/10^3 < 5$ like in the dry cases. The difference of the decrease of F_n^* in the lubricated cases is due to the presence of fluid particles in the gap between the substrate and the indenter. These fluid particles are squeezed out during the starting phase of the scratching [40]. The normal force on the indenter exhibits stronger fluctuations compared to the tangential force during the scratching.

The coefficient of friction μ was calculated from the tangential and normal force and therefore shows similar characteristics. In the dry cases, the coefficient of friction μ increases within $L_{scr}^* < 10$ up to $\mu \approx 0.6$ and then fluctuates in a range between 0.5 and 1.2. The differences between the individual simulations build up to a scratching length of $L_{scr}^* = 80$. As already observed for the normal force, the starting phase in which the coefficient of friction increases from $\mu = 0$ at $L_{scr}^* = 0$ is longer in the lubricated cases. Up to a scratching length of $L_{scr}^* \approx 20$, the coefficient of friction increases and fluctuates in the range $0.5 < \mu < 1.2$ for $L_{scr}^* > 20$.

For a better comparison, the simulation data were averaged in the quasi steady state regime. This regime is defined as $L_{scr}^* > 20$. It has been shown that the number of fluid particles in the gap between the indenter and the substrate is practically constant in this

steady state regime [40]. Fig. 4 shows the number of fluid particles in the gap between the indenter and the substrate. Starting at the beginning of the scratching, the fluid particles are squeezed out of the gap. At the beginning of the scratching, there are around 90 particles in the gap that are rapidly squeezed out of the gap with ongoing lateral movement. In the simulations with higher scratching velocity, there is a tendency that the particles are squeezed out slightly faster – especially from $L^*_{\mathrm{scr}} = 10$ up to $L^*_{\mathrm{scr}} = 40$. Until $L^*_{\mathrm{scr}} = 20$, the number of particles remaining in the gap is reduced to less than 20 particles in all simulations. These few isolated atoms are scattered in the contact zone. Consequently, the actual contact zone is mainly dry and there is no fluid flow in the contact zone. For $L^*_{\mathrm{scr}} > 20$, the number of particles in the gap remains approximately constant. This establishes the steady state regime. The results for the friction forces and the coefficient of friction (cf. Fig. 3) corroborate this classification.

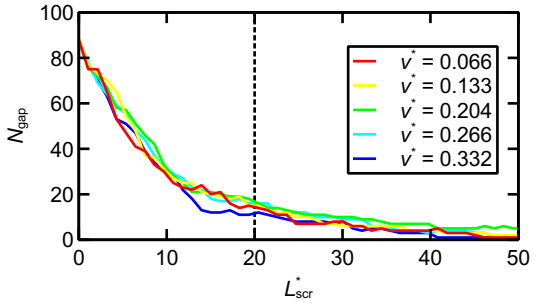


Figure 4 Number of fluid particles in the gap N_{gap} between substrate and indenter as function of scratching length L^*_{scr} in the lubricated cases (cf. Fig. 2).

Fig. 5 shows the mean coefficient of friction μ as a function of the scratching velocity v^* averaged in the steady state regime. No distinct dependency of the coefficient of friction on the scratching velocity is found in the considered velocity range. For a scratching velocity $v^* = 0.066$ the coefficients of friction in the lubricated and the dry case are almost equal and lie around $\mu \approx 0.85$. Nearly equal values for the coefficient of friction at this velocity have also been observed in a previous work of our group [38] with a similar simulation scenario. Therein, a cylindrical indenter with a slightly smaller indenter radius was used. Furthermore, for the simulations reported in Ref. [38] no thermostats were applied in the indenter and the fluid. For the higher scratching velocities, the coefficient of friction in the dry and the lubricated cases remains nearly constant, but scatters intensely. The values fluctuate in the range $0.7 < \mu < 0.9$ for the dry and the lubricated cases. For moderate scratching velocities ($0.1 < v^* < 0.3$), the lubricated cases yield a slightly lower coefficient of friction. Nevertheless, these differences are probably within the uncertainty of the data. Noreyan and Amar [32] and Zhu et al. [50] have performed dry scratching simulations at different scratching velocities and found a decrease of the coefficient of friction by at least 25 % with increasing scratching velocity in the range $v^* = 0.16$ – $v^* = 1.42$. This decrease was reported [32, 50] to be more pronounced at high scratching velocities, i.e. $v^* > 0.332$ (which corresponds to approximately $v > 100$ m/s). However, it should be noted that the simulation scenarios from Refs. [32, 50] and from this work differ regarding the indenter shape and the molecular force fields which is known to influence the coefficient of friction [2]. For indentation processes, it has been found that the effect of the indenter velocity on the simulation results is small as long as it is below around 100 m/s [3, 17, 36]. The reason hereto is that the plastic deformation processes in the sample are based on dislocation generation, which are only little affected by the indenter

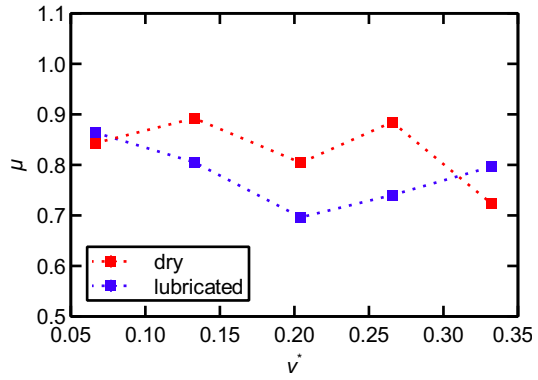


Figure 5 Mean coefficient of friction μ as function of the scratching velocity v^* for the dry and the lubricated cases. Lines are a guide to the eye.

velocity in the regime of around 0.01 – 100 m/s. At smaller indenter velocities, however, thermal effects may influence dislocation generation, reaction and migration, while at higher velocities, the sample starts to amorphize locally.

Fig. 6 shows the work done by the indenter W_I^* as function of the scratching velocity v^*. All values lie in the range $2.1 < W_\mathrm{I}^*/10^6 < 2.4$. For both the dry and the lubricated case,

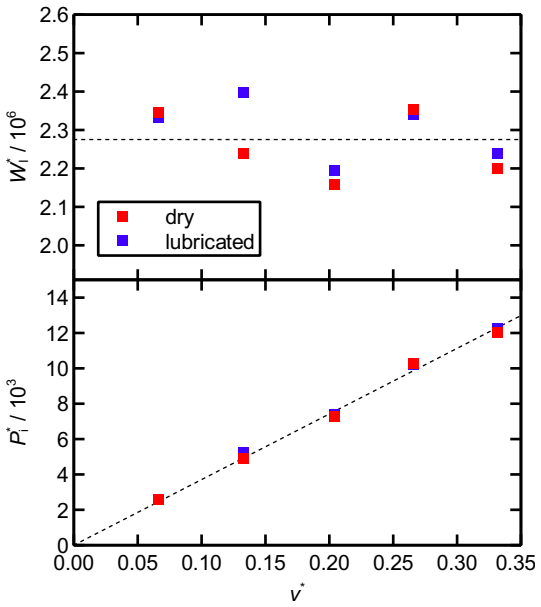


Figure 6 Work done by the indenter W_I^* (top) and power supplied to the system by the indenter P_I^* in the steady state regime as function of the scratching velocity v^* for the dry and the lubricated cases. The dotted lines indicate the average of the work done by the indenter $\overline{W_\mathrm{I}^*} = 2.28 \times 10^6$ (top) of all simulations and a linear function (bottom) $P_\mathrm{I}^* = a \cdot v^*$ fitted to the data of the dry simulations ($a = 37.1 \times 10^3$).

no dependency on the scratching velocity is observed as expected from the results for the tangential force (cf. Fig. 3). The power P_I^* supplied to the system by the indenter motion is also shown in Fig. 6. The power P_I^* in both the dry and the lubricated case is linearly dependent on the scratching velocity v^* as expected from the nearly constant work done

by the indenter shown in Fig. 6 and the shortening of the simulation time with increasing scratching velocity. The values of the power reach from $P_{\mathrm{I}}^{*} = 2.6 \times 10^{3}$ at $v^{*} = 0.066$ to $P_{\mathrm{I}}^{*} = 12.2 \times 10^{3}$ at $v^{*} = 0.332$. No significant differences between the dry and the lubricated cases are observed. The dotted line shown in Fig. 6 represents the linear function $P_{\mathrm{I}}^{*} = 37.1 \times 10^{3} \cdot v^{*}$ that was fitted to the results of the dry cases. The form of the function was chosen such that it reflects the limit $P_{\mathrm{I}}^{*} = 0$ for $v^{*} \to 0$. For $v^{*} \to 0$, the scratching time becomes infinite and the power goes to zero.

3.2 Thermal Properties

Besides the mechanical properties, also thermal properties of a contact process play an important role in technical applications. First, the change of the energy ΔE_{j}^{*} in the three parts of the simulation (substrate, indenter and fluid) are shown in Fig. 7 for the steady state regime ($L_{\mathrm{scr}}^{*} > 20$). The sampled changes of the energies are found to be independent of the scratching velocity. The work done by the indenter is portioned into a change of the energy of the substrate, of the fluid and of the indenter. The largest portion of the energy supplied to the system by the indenter goes into the substrate with about $\overline{\Delta E_{\mathrm{S}}^{*}} = 2.1 \times 10^{6}$ in the dry cases and $\overline{\Delta E_{\mathrm{S}}^{*}} = 1.94 \times 10^{6}$ in the lubricated cases. The increase of the energy of the indenter $\Delta E_{\mathrm{I}}^{*}$ are much smaller than the increase of the energy of the substrate $\Delta E_{\mathrm{S}}^{*}$ and are around $0.08 < \Delta E_{\mathrm{I}}^{*}/10^{6} < 0.16$. The change of the energy of the fluid $\Delta E_{\mathrm{F}}^{*}$ is about twice the change of the energy of the indenter $\Delta E_{\mathrm{I}}^{*}$ in the lubricated cases and lies in the range $0.2 < \Delta E_{\mathrm{F}}^{*}/10^{6} < 0.25$. The energy increase of the substrate and the indenter is significantly reduced in the lubricated cases compared to the dry ones. This reduction is simply due to the fact that a significant portion of the work done by the indenter goes into the fluid – if such is present. The reduction of the energy increase of the substrate and the indenter is fairly independent of the scratching velocity.

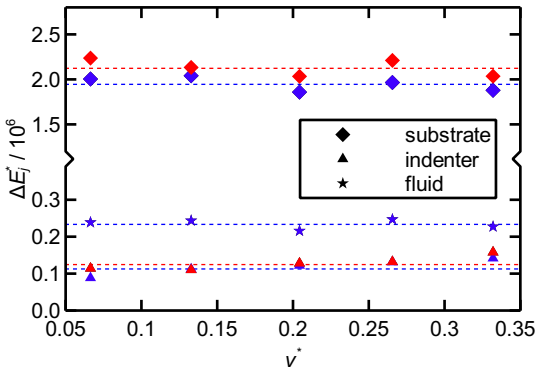


Figure 7 Change of the energy in the substrate $\Delta E_{\mathrm{S}}^{*}$, the indenter $\Delta E_{\mathrm{I}}^{*}$ and the fluid $\Delta E_{\mathrm{F}}^{*}$ in the steady state regime ($L_{\mathrm{scr}}^{*} > 20$) as function of the scratching velocity v^{*} in the dry (red) and the lubricated (blue) cases. The dotted lines represent the mean over all scratching velocities of the respective data (dry: $\overline{\Delta E_{\mathrm{S}}^{*}} = 2.12 \times 10^{6}$, $\overline{\Delta E_{\mathrm{I}}^{*}} = 0.12 \times 10^{6}$; lubricated: $\overline{\Delta E_{\mathrm{S}}^{*}} = 1.94 \times 10^{6}$, $\overline{\Delta E_{\mathrm{I}}^{*}} = 0.11 \times 10^{6}$, $\overline{\Delta E_{\mathrm{F}}^{*}} = 0.23 \times 10^{6}$).

To evaluate the energy balance in more detail, we compare the sum of the means over all scratching velocities of the changes of the energy to the work done by the indenter using the following equation: $\overline{W_{\mathrm{I}}^{*}} = \overline{\Delta E_{\mathrm{S}}^{*}} + \overline{\Delta E_{\mathrm{I}}^{*}} + \overline{\Delta E_{\mathrm{F}}^{*}} + \delta$. Here, δ represents the numerical error which arises from rounding errors as well as the error occurring during numerical integration. The error is less than 1 % in all simulations. The energy balance reveals that in the dry

cases around 94.2 % of the work done by the indenter causes an increase of the energy of the substrate. The remainder of the work done by the indenter leads to an increase of the energy of the indenter. In the lubricated cases, 10 % of the work done by the indenter causes an increase of the energy of the fluid. Therefore, less energy goes into the substrate (about 84 %) and into the indenter (only about 5 %). These values are similar to results obtained in a previous work of our group [38]. Comparing the dry and the lubricated cases, the energy increase of the substrate ΔE^*_S is reduced by 8.5 % and the energy increase ΔE^*_I of the indenter is reduced by 9.5 %. This is due to the increase of the energy of the fluid ΔE^*_F and leads to the expected cooling effect for both the substrate and the indenter.

The results obtained from the simulations with different scratching velocities for the change of the energy per time $\mathrm{d}E^*_j/\mathrm{d}t^*$ in the steady state regime ($L^*_\mathrm{scr} > 20$) are shown in Fig. 8. The numbers obtained from $\mathrm{d}E^*_j/\mathrm{d}t^*$ for the substrate, the fluid and the indenter show a linear dependency on the scratching velocity. This is expected as the power supplied to the system also increases linearly with increasing scratching velocity. As expected from the changes of the energy ΔE^*_j shown in Fig. 7, the values $\mathrm{d}E^*_j/\mathrm{d}t^*$ of the substrate ($j = \mathrm{S}$) are highest for all simulations (cf. Fig. 8). This is probably due to the fact that the main part of the work done by the indenter dissipates in the substrate in the vicinity of the contact as a consequence of elastic and plastic deformation.

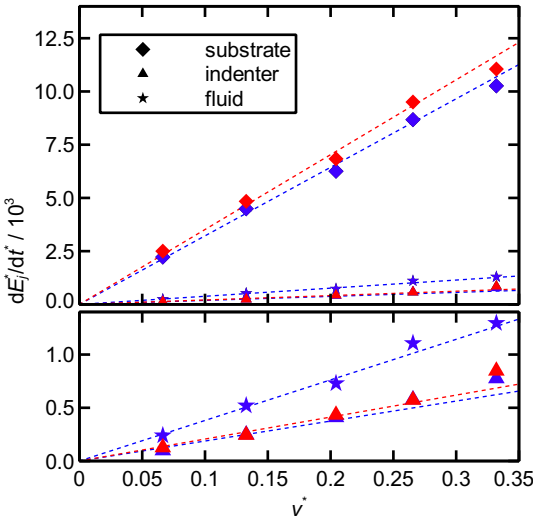


Figure 8 Change of the energy per time in the substrate $\mathrm{d}E^*_\mathrm{S}/\mathrm{d}t^*$, the indenter $\mathrm{d}E^*_\mathrm{I}/\mathrm{d}t^*$ and the fluid $\mathrm{d}E^*_\mathrm{F}/\mathrm{d}t^*$ as function of the scratching velocity v^* in the dry (red) and the lubricated (blue) cases. The dotted lines represent linear functions of the form $\mathrm{d}E^*_j/\mathrm{d}t^* = a \cdot v^*$ fitted to the respective data (dry: $a_\mathrm{S}/10^3 = 35.2$, $a_\mathrm{I}/10^3 = 2.1$; lubricated: $a_\mathrm{S}/10^3 = 32.2$, $a_\mathrm{I}/10^3 = 1.9$, $a_\mathrm{F}/10^3 = 3.8$).

Fig. 9 shows the temperature of the contact zone of the substrate as a function of the scratching velocity. The contact zone temperature was averaged both spatially (cf. Fig. 2) and over time; during the steady state regime. The results of the simulations with a scratching velocity of $v^* = 0.066$ show that the contact zone temperature ($T^*_\mathrm{contact} \approx 0.85$) is slightly increased compared to the initial temperature of $T^* = 0.8$, which was also applied to the thermostats during the contact process. With increasing scratching velocity, the contact zone temperature increases linearly. This is a consequence of the dependency of the change of the energy per time on the scratching velocity, cf. Fig. 8. The contact zone temperature can be described by a linear function which assumes no temperature change of the contact zone

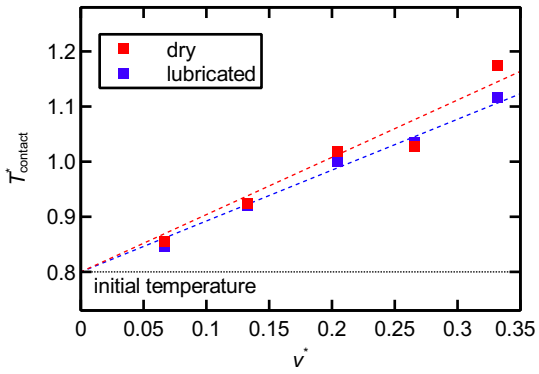


Figure 9 Temperature of the substrate in the contact zone T^*_{contact} (cf. Fig. 2) as function of the scratching velocity v^* in the dry and the lubricated cases averaged in the steady state regime ($L^*_{\text{scr}} > 20$). The black dotted line indicates the preset temperature of the thermostats. The red and blue dotted lines represent linear functions $T^*_{\text{contact}} = a \cdot v^* + 0.8$ fitted to the respective data (dry: $a = 1.04$; lubricated: $a = 0.92$).

in the limit $v^* \to 0$. For most scratching velocities, the contact zone temperature is higher in the dry cases than in the corresponding lubricated cases (except the scratching velocity of $v^* = 0.266$). This effect is due to the presence of the fluid since a part of the work done by the indenter goes into the fluid (cf. Fig. 8), which confirms the findings from Ref. [38]. The cooling of the substrate in the contact zone is the primary task of the fluid in macroscopic cutting processes [8, 9, 22]. The absolute difference of the contact zone temperature between the dry and the lubricated cases slightly increases with increasing scratching velocity. This is also in accordance to the increase of the change of the energy per time of the fluid with increasing scratching velocity (cf. Fig. 8). The increase of the contact zone temperature with increasing scratching velocity has already been reported by several studies [19, 29, 31, 48, 50] for dry cases. The results from the present study extend this phenomenon to lubricated systems, where the thermal balance is influenced by several additional phenomena at the interface and in the fluid [38].

The spatial temperature distribution is discussed by means of snapshots of the x^*-z^* plane at the scratching length of $L^*_{\text{scr}} = 77$, cf. Fig. 10. The setup of the bins used for the spatial discretization is shown in Fig. 2. Surface lines of the substrate and the indenter are directly extracted from the position data of the particles using an alpha shape algorithm. Details of the applied post-processing algorithm can be found in Ref. [14]. The snapshots are shown for the scratching velocities $v^* = 0.066$, 0.204 and 0.332. For the velocity $v^* = 0.066$, there is a slight increase of the local temperature up to $T^* \approx 0.96$ observed in the tip of the chip in the dry case. In the bulk of the substrate, no significant change of the temperature compared to the initial temperature ($T^* = 0.8$) is found. In the lubricated cases, the increase of the local temperature in the contact zone is reduced by the presence of the fluid (cf. Fig. 10). Increasing the scratching velocity yields an increase of the temperature in the chip in the dry cases (cf. Fig. 10 at $v^* = 0.204$ and 0.332). The maximum local temperature at the velocity of $v^* = 0.332$ is $T^* = 1.77$ and is reached in the chip. Overall the warming of the substrate is mainly limited to the chip and the direct vicinity of the contact zone, which is the most plastically deformed zone of the substrate. The concentration of warming of the substrate in the chip has also been reported by Refs. [11, 28]. For the temperature field of the fluid in the lubricated cases, the temperature is mainly increased in the area around the chip. This indicates that the increase of the internal energy of the fluid mainly takes place by a heat flux from the chip. Therefore, only a small portion of the increase of the internal

energy of the fluid results from actual energy dissipation in the fluid in the contact zone, as practically no fluid is present there. The dissipated energy which increases the internal energy of the fluid during the scratching is hence mainly indirectly transferred to the fluid via the chip and the substrate surface (cf. Fig. 4). The warming of the fluid includes larger spatial dimensions compared to the substrate, which is a direct result of the differences of the thermal conductivity of the substrate, the indenter and the fluid [38]. The fluid has a significantly lower thermal conductivity than the substrate [10, 44].

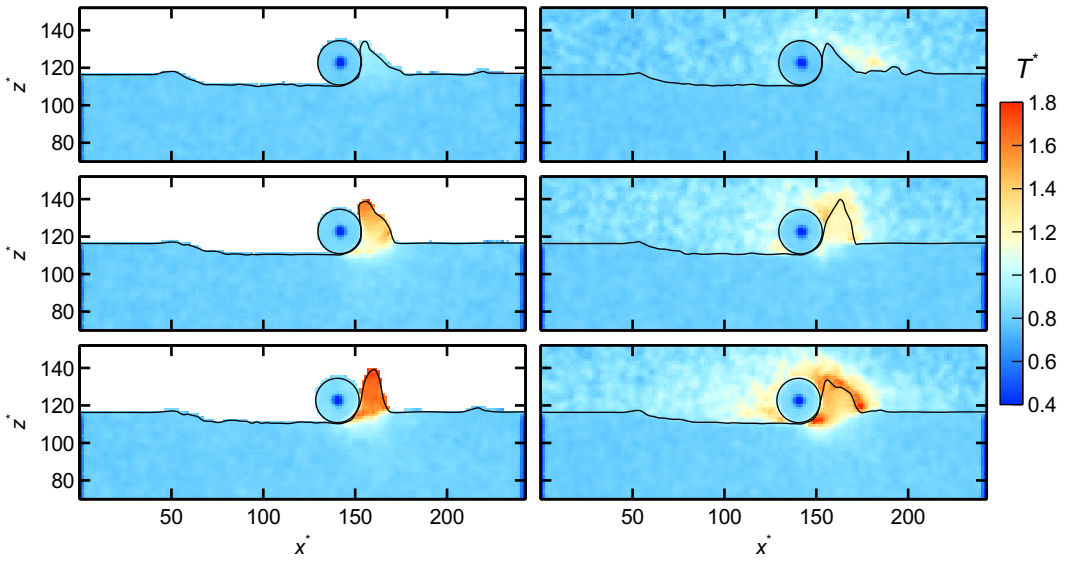


Figure 10 Temperature in x^*-z^* plane for the scratching length $L^*_{\mathrm{scr}} = 77$ for dry (left) and lubricated (right) cases for the scratching velocities $v^* = 0.066$ (top), 0.204 (middle) and 0.332 (bottom). Black lines indicate the surface of the substrate and the indenter.

4 Conclusions

In the present work, the influence of the scratching velocity on dry and lubricated contact processes was studied by molecular dynamics simulations. All intermolecular interactions were modeled by the LJTS potential such that the model is a simplified yet representative system. Mechanical properties like the occurring forces, the coefficient of friction and the power supplied to the system as well as thermal properties like the change of the energy, the energy balance and the local temperature field were evaluated.

The mechanical properties show only a weak dependency on the scratching velocity in the considered velocity range. No significant dependency on the scratching velocity was observed for the coefficient of friction in both dry and lubricated cases. The power supplied to the system by the indenter increases linearly with increasing scratching velocity, which is expected from the shortening of the total scratching time while the total energy supplied to the system remains fairly constant. This independence can be traced back to the plastic deformation processes in the material, which do not change in the velocity regime considered here. The increase of the power supplied to the system by the indenter directly affects the temperature in the contact zone which also shows a linear increase with increasing scratching velocity both in the dry and in the lubricated system. In the lubricated cases, the local temperature in the vicinity of the contact zone is slightly reduced due to the presence of the fluid. The cooling effect of the lubricant is weaker than could have been expected from macroscopic experience. This could indicate that the macroscopic cooling effect of the lubricant is mainly

indirect, i.e. due to a cooling of the substrate and the indenter not in the vicinity of the contact zone but in its surroundings, leading to an increased removal of heat by conduction in the macroscopic system.

References

1 A. T. Al Motasem, J. Bergström, A. Gåård, P. Krakhmalev, and L. J. Holleboom. Atomistic insights on the wear/friction behavior of nanocrystalline ferrite during nanoscratching as revealed by molecular dynamics. *Tribology Letters*, 65(3):101, 2017. doi:10.1007/s11249-017-0876-y.

2 I. Alabd Alhafez, A. Brodyanski, M. Kopnarski, and H. M. Urbassek. Influence of tip geometry on nanoscratching. *Tribology Letters*, 65(1):26, 2017. doi:10.1007/s11249-016-0804-6.

3 J. Alcalá, R. Dalmau, O. Franke, M. Biener, J. Biener, and A. Hodge. Planar defect nucleation and annihilation mechanisms in nanocontact plasticity of metal surfaces. *Physical Review Letters*, 109(7):075502, 2012. doi:10.1103/PhysRevLett.109.075502.

4 M. P. Allen and D. J. Tildesley. *Computer simulation of liquids*. Oxford University Press, Oxford, United Kingdom, second edition edition, 2017.

5 S. Becker, H. M. Urbassek, M. T. Horsch, and H. Hasse. Contact angle of sessile drops in Lennard-Jones systems. *Langmuir*, 30(45):13606–13614, 2014. doi:10.1021/la503974z.

6 H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *The Journal of Chemical Physics*, 81(8):3684–3690, 1984. doi:10.1063/1.448118.

7 E. Brinksmeier, J. C. Aurich, E. Govekar, C. Heinzel, H.-W. Hoffmeister, F. Klocke, J. Peters, R. Rentsch, D. J. Stephenson, E. Uhlmann, K. Weinert, and M. Wittmann. Advances in modeling and simulation of grinding processes. *CIRP Annals*, 55(2):667–696, 2006. doi:10.1016/j.cirp.2006.10.003.

8 E. Brinksmeier, C. Heinzel, and M. Wittmann. Friction, cooling and lubrication in grinding. *CIRP Annals*, 48(2):581–598, 1999. doi:10.1016/S0007-8506(07)63236-3.

9 E. Brinksmeier, D. Meyer, A. G. Huesmann-Cordes, and C. Herrmann. Metalworking fluids–mechanisms and performance. *CIRP Annals*, 64(2):605–628, 2015. doi:10.1016/j.cirp.2015.05.003.

10 M. Bugel and G. Galliero. Thermal conductivity of the Lennard-Jones fluid: An empirical correlation. *Chemical Physics*, 352(1-3):249–257, 2008. doi:10.1016/j.chemphys.2008.06.013.

11 Y. H. Chen, H. Han, F. Z. Fang, and X. T. Hu. MD simulation of nanometric cutting of copper with and without water lubrication. *Science China Technological Sciences*, 57(6):1154–1159, 2014. doi:10.1007/s11431-014-5519-z.

12 F. Diewald, M. P. Lautenschlaeger, S. Stephan, K. Langenbach, C. Kuhn, S. Seckler, H.-J. Bungartz, H. Hasse, and R. Müller. Molecular dynamics and phase field simulations of droplets on surfaces with wettability gradient. *Computer Methods in Applied Mechanics and Engineering*, 361:112773, 2020. doi:10.1016/j.cma.2019.112773.

13 Y. Dong, Q. Li, and A. Martini. Molecular dynamics simulation of atomic friction: A review and guide. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, 31(3):030801, 2013. doi:10.1116/1.4794357.

14 H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–559, 1983. doi:10.1109/TIT.1983.1056714.

15 J. P. Ewen, D. M. Heyes, and D. Dini. Advances in nonequilibrium molecular dynamics simulations of lubricants and additives. *Friction*, 6(4):349–386, 2018. doi:10.1007/s40544-018-0207-9.

16 Y. Gao, A. Brodyanski, M. Kopnarski, and H. M. Urbassek. Nanoscratching of iron: A molecular dynamics study of the influence of surface orientation and scratching direction. *Computational Materials Science*, 103:77–89, 2015. doi:10.1016/j.commatsci.2015.03.011.

17 Y. Gao, C. J. Ruestes, D. R. Tramontina, and H. M. Urbassek. Comparative simulation study of the structure of the plastic zone produced by nanoindentation. *Journal of the Mechanics and Physics of Solids*, 75:58–75, 2015. doi:10.1016/j.jmps.2014.11.005.

18 Y. Gao and H. M. Urbassek. Evolution of plasticity in nanometric cutting of Fe single crystals. *Applied Surface Science*, 317:6–10, 2014. doi:10.1016/j.apsusc.2014.08.020.

19 Y. B. Guo, Y. C. Liang, M. J. Chen, Q. S. Bai, and L. H. Lu. Molecular dynamics simulations of thermal effects in nanometric cutting process. *Science China Technological Sciences*, 53(3):870–874, 2010. doi:10.1007/s11431-009-0243-9.

20 T. Halicioğlu and G. M. Pound. Calculation of potential energy parameters from crystalline state properties. *Physica Status Solidi (a)*, 30(2):619–623, 1975. doi:10.1002/pssa.2210300223.

21 M. Heier, S. Stephan, J. Liu, W. G. Chapman, H. Hasse, and K. Langenbach. Equation of state for the Lennard-Jones truncated and shifted fluid with a cut-off radius of 2.5 σ based on perturbation theory and its applications to interfacial thermodynamics. *Molecular Physics*, 116(15-16):2083–2094, 2018. doi:10.1080/00268976.2018.1447153.

22 C. Heinzel, B. Kirsch, D. Meyer, and J. Webster. Interactions of grinding tool and supplied fluid. *CIRP Annals*, 69(1):624–645, 2020. doi:10.1016/j.cirp.2020.05.001.

23 Y.-L. Huang, T. Merker, M. Heilig, H. Hasse, and J. Vrabec. Molecular modeling and simulation of vapor–liquid equilibria of ethylene oxide, ethylene glycol, and water as well as their binary mixtures. *Industrial & Engineering Chemistry Research*, 51(21):7428–7440, 2012. doi:10.1021/ie300248z.

24 B. Kirsch, S. Basten, H. Hasse, and J. C. Aurich. Sub-zero cooling: A novel strategy for high performance cutting. *CIRP Annals*, 67(1):95–98, 2018. doi:10.1016/j.cirp.2018.04.060.

25 F. Klocke. *Fertigungsverfahren 2: Zerspanung mit geometrisch unbestimmter Schneide*. VDI-Buch. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017. doi:10.1007/978-3-662-53310-9.

26 M. P. Lautenschlaeger and H. Hasse. Transport properties of the Lennard-Jones truncated and shifted fluid from non-equilibrium molecular dynamics simulations. *Fluid Phase Equilibria*, 482:38–47, 2019. doi:10.1016/j.fluid.2018.10.019.

27 M. P. Lautenschlaeger, S. Stephan, M. T. Horsch, B. Kirsch, J. C. Aurich, and H. Hasse. Effects of lubrication on friction and heat transfer in machining processes on the nanoscale: A molecular dynamics approach. *Procedia CIRP*, 67:296–301, 2018. doi:10.1016/j.procir.2017.12.216.

28 M. P. Lautenschlaeger, S. Stephan, H. M. Urbassek, B. Kirsch, J. C. Aurich, M. T. Horsch, and H. Hasse. Effects of lubrication on the friction in nanometric machining processes: A molecular dynamics approach. *Applied Mechanics and Materials*, 869:85–93, 2017. doi:10.4028/www.scientific.net/AMM.869.85.

29 Y. Liu, B. Li, and L. Kong. A molecular dynamics investigation into nanoscale scratching mechanism of polycrystalline silicon carbide. *Computational Materials Science*, 148:76–86, 2018. doi:10.1016/j.commatsci.2018.02.038.

30 B. Magyar and B. Sauer. Methods for the simulation of the pressure, stress, and temperature distribution in the contact of fractal generated rough surfaces. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 231(4):489–502, 2017. doi:10.1177/1350650115593962.

31 B. Meng, D. Yuan, and S. Xu. Study on strain rate and heat effect on the removal mechanism of SiC during nano-scratching process by molecular dynamics simulation. *International Journal of Mechanical Sciences*, 151:724–732, 2019. doi:10.1016/j.ijmecsci.2018.12.022.

32 A. Noreyan and J. G. Amar. Molecular dynamics simulations of nanoscratching of 3C SiC. *Wear*, 265(7):956–962, 2008. doi:10.1016/j.wear.2008.02.020.

33 S. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1):1–19, 1995. doi:10.1006/jcph.1995.1039.

34 J. Ren, J. Zhao, Z. Dong, and P. Liu. Molecular dynamics study on the mechanism of AFM-based nanoscratching process with water-layer lubrication. *Applied Surface Science*, 346:84–98, 2015. doi:10.1016/j.apsusc.2015.03.177.

35 R. Rentsch and I. Inasaki. Effects of fluids on the surface generation in material removal processes –molecular dynamics simulation–. *CIRP Annals*, 55(1):601–604, 2006. doi:10.1016/S0007-8506(07)60492-2.

36 C. J. Ruestes, E. M. Bringa, Y. Gao, and H. M. Urbassek. Molecular dynamics modeling of nanoindentation. In *Applied nanoindentation in advanced materials*, pages 313–345. John Wiley & Sons, Ltd, 2017. doi:10.1002/9781119084501.ch14.

37 A. J. Schultz and D. A. Kofke. Comprehensive high-precision high-accuracy equation of state and coexistence properties for classical Lennard-Jones crystals and low-temperature fluid phases. *The Journal of Chemical Physics*, 149(20):204508, 2018. doi:10.1063/1.5053714.

38 S. Stephan, M. Dyga, H. M. Urbassek, and H. Hasse. The influence of lubrication and the solid–fluid interaction on thermodynamic properties in a nanoscopic scratching process. *Langmuir*, 35(51):16948–16960, 2019. doi:10.1021/acs.langmuir.9b01033.

39 S. Stephan, M. T. Horsch, J. Vrabec, and H. Hasse. MolMod – an open access database of force fields for molecular simulations of fluids. *Molecular Simulation*, 45(10):806–814, 2019. doi:10.1080/08927022.2019.1601191.

40 S. Stephan, M. P. Lautenschlaeger, I. A. Alhafez, M. T. Horsch, H. M. Urbassek, and H. Hasse. Molecular dynamics simulation study of mechanical effects of lubrication on a nanoscale contact process. *Tribology Letters*, 66(4):126, 2018. doi:10.1007/s11249-018-1076-0.

41 S. Stephan, M. Thol, J. Vrabec, and H. Hasse. Thermophysical properties of the Lennard-Jones fluid: Database and data assessment. *Journal of Chemical Information and Modeling*, 59(10):4248–4265, 2019. doi:10.1021/acs.jcim.9b00620.

42 I. Szlufarska, M. Chandross, and R. W. Carpick. Recent advances in single-asperity nanotribology. *Journal of Physics D: Applied Physics*, 41(12):123001, 2008. doi:10.1088/0022-3727/41/12/123001.

43 A. Travesset. Phase diagram of power law and Lennard-Jones systems: Crystal phases. *The Journal of Chemical Physics*, 141(16):164501, 2014. doi:10.1063/1.4898371.

44 K. V. Tretiakov and S. Scandolo. Thermal conductivity of solid argon from molecular dynamics simulations. *The Journal of Chemical Physics*, 120(8):3765–3769, 2004. doi:10.1063/1.1642611.

45 A. I. Vakis, V. A. Yastrebov, J. Scheibert, L. Nicola, D. Dini, C. Minfray, A. Almqvist, M. Paggi, S. Lee, G. Limbert, J. F. Molinari, G. Anciaux, R. Aghababaei, S. Echeverri Restrepo, A. Papangelo, A. Cammarata, P. Nicolini, C. Putignano, G. Carbone, S. Stupkiewicz, J. Lengiewicz, G. Costagliola, F. Bosia, R. Guarino, N. M. Pugno, M. H. Müser, and M. Ciavarella. Modeling and simulation in tribology across scales: An overview. *Tribology International*, 125:169–199, 2018. doi:10.1016/j.triboint.2018.02.005.

46 J. Vrabec, G. K. Kedia, G. Fuchs, and H. Hasse. Comprehensive study of the vapour–liquid coexistence of the truncated and shifted Lennard–Jones fluid including planar and spherical interface properties. *Molecular Physics*, 104(9):1509–1527, 2006. doi:10.1080/00268970600556774.

47 P. Wang, J. Yu, and Q. Zhang. Nano-cutting mechanical properties and microstructure evolution mechanism of amorphous/single crystal alloy interface. *Computational Materials Science*, 184:109915, 2020. doi:10.1016/j.commatsci.2020.109915.

48 J. Zhang, T. Sun, Y. Yan, and Y. Liang. Molecular dynamics study of scratching velocity dependency in AFM-based nanometric scratching process. *Materials Science and Engineering: A*, 505(1):65–69, 2009. doi:10.1016/j.msea.2008.10.049.

49 Z. Zhang, I. Alabd Alhafez, and H. M. Urbassek. Scratching an Al/Si interface: Molecular dynamics study of a composite material. *Tribology Letters*, 66(3):86, 2018. doi:10.1007/s11249-018-1038-6.

50 P.-Z. Zhu, C. Qiu, F.-Z. Fang, D.-D. Yuan, and X.-C. Shen. Molecular dynamics simulations of nanometric cutting mechanisms of amorphous alloy. *Applied Surface Science*, 317:432–442, 2014. doi:10.1016/j.apsusc.2014.08.031.

The Discrete Morse Complex of Images: Algorithms, Modeling and Applications

Ricardo Dutra da Silva
Department of Informatics, Federal University of Technology, Curitiba, PR, Brazil

Helio Pedrini
Institute of Computing, University of Campinas, SP, Brazil

Bernd Hamann
Department of Computer Science, University of California, Davis, CA 95616, USA

Abstract

The Morse complex can be used for studying the topology of a function, e.g., an image or terrain height field when understood as bivariate functions. We present an algorithm for the computation of the discrete Morse complex of two-dimensional images using an edge-based data structure. By using this data structure, it is possible to perform local operations efficiently, which is important to construct the complex and make the structure useful for areas like visualization, persistent homology computation, or construction of a topological hierarchy. We present theoretical and applied results to demonstrate benefits and use of our method.

2012 ACM Subject Classification Computing methodologies → Image processing

Keywords and phrases Discrete Morse Complex, Image Topology, Cell Complexes

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.18

1 Introduction

Morse theory [37] is primarily used for the study of a function's topology by establishing the topological relationships between its critical points, i.e., its extrema and saddles [18]. Algorithms and models for Morse complexes have become increasingly important as fundamental building blocks of computational topology with applications in image analysis, e.g., for image segmentation, object skeletonization and classification. Relevant literature for Morse-Smale complexes includes [8, 27, 42], discrete Morse complexes are covered in [17, 30, 50] and persistent homology is discussed in [11, 15, 19, 20, 33, 50].

In many image processing problems, functions represent a space with scalar measures over it. This is, for instance, the case of height fields (terrains). The domain of such functions is sampled and, therefore, the continuous Morse theory is not directly applied. The adaptation of the continuous theory to sampled data has been studied in [9, 18, 53], for example, where the authors use the simulation of smooth notions to guide the computation and produce what is called a Morse-Smale complex [53].

The discrete Morse theory, formulated by Forman [21, 22], is another adaptation of the Morse theory to discrete structures. It was explored in works such as King et al. [31] and Lewiner et al. [34]. Robins et al. [44] provided algorithms for computing discrete Morse complexes through an image analysis approach.

Our contribution is an algorithm- and data structure-driven approach to compute the discrete Morse complex of two-dimensional images. The complex is similar to the Morse-Smale [9, 18, 26, 45, 53] approach and the method described by Robins et al. [44]. We introduce a specialization for the specific case of 2D image functions of the approach described by Robins et al. [44] and show that the specialized algorithms are optimized for the case where paths emanate from saddles, instead of using the breadth-first search discussed in [44]. We contribute a complete, detailed description of an algorithm to extract paths

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 18; pp. 18:1–18:19

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

from a vector field given as input, resulting in the discrete Morse complex of an image. The idea is similar to the one presented by Edelsbrunner et al. [18] for Morse-Smale complexes; we differ by using the discrete approach of [44]. In addition, we show that the complexes can be easily implemented by means of an edge-based data structure. Furthermore, since we use an edge-based data structure, the complex is suitable for computing topological persistence [6, 10, 12, 14, 35, 40, 51] and topological hierarchies [43, 49], as well as for visualization purposes.

In Sections 2 to 4, we present basic concepts on topology of images, discrete Morse complexes and data structures. These concepts are used in the algorithms and in the modeling of the resulting complexes, which are discussed in detail in Sections 5 to 7. In Section 8, we show the application of the discrete Morse complex to compute the persistence and hierarchical representations of images. Finally, we present some concluding remarks in Section 9.

2 Images as Cell Complexes and Topological Data Structures

A p-cell, of dimension p, is the basic element to construct the discrete domain which is a cell complex. It is defined as a topological space homeomorphic to a p-ball $B^p = \{x \in \mathbb{R}^p \colon |x| \leq 1\}$. The first three low-order p-cells are 0-cells (also known as nodes), 1-cells (edges) and 2-cells (faces). Even though the secondary names are commonly used in computer science, the concept of a face is a different one in topology, as we will state soon. Therefore, to be consistent with the topology and computational topology literature, we choose to use the primary names. We will denote a p-cell as α^p or simply α whenever the dimension is clear from the context.

The boundary of a p-cell consists of cells of lower dimension that limit the cell. A face of a p-cell σ^p is a cell τ^k, with $k \leq p$, which is part of the boundary of the p-cell. The p-cell σ^p is called a coface of τ^k. The bounding relations of face and coface will be stated as $\tau^k \prec \sigma^p$ and $\sigma^p \succ \tau^k$.

A cell complex K is a finite collection of cells that satisfies the following requirements: (1) all the faces of a cell in the complex also belong to the complex, and (2) the intersection of any two cells is either empty or a face of both cells. A p-complex is a cell complex such that all its cells have dimension less than or equal to p. A subcomplex of a cell complex K is a subset of cells $L \subseteq K$ such that L is also a cell complex.

An image is viewed as a function $f \colon D \to \mathbb{R}$, defined on a subset of the discrete lattice, $D = \{(x, y) \in \mathbb{Z}^2 \mid 1 \leq x \leq M, 1 \leq y \leq N\}$. A point in D, and its value, is called a pixel. An image can be modeled by a regular two-dimensional cell complex K, as described by Kovalevsky [32]. Kovalevsky adopts a model where pixels are 2-cells since both are area-related elements. The dual of this representation is also commonly used, for example, in Robins et al. [44]. We adopt this latter model, such that an image is a regular complex K with 0-cells corresponding to the pixels in D. The 2-cells are usually squares or triangles defined by a pixel and some of its closest pixels in D. The 1-cells are faces of the 2-cells. An example of such a model is shown in Figure 1.

Different data structures can be used to model cell complexes [16, 20, 24, 27, 52]. Since we are interested in two-dimensional cell complexes, planar edge-based data structures, such as the half-edge or the quad-edge, are efficient for our purposes [5, 25, 36]. As we will show in our algorithms, efficient computation can be performed through an edge-based data structure, since it allows adjacency operations to be computed in constant time. Hereafter, we will assume that the complexes are all modeled by means of an edge-based data structure.

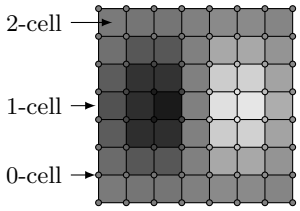


Figure 1 Digital image modeled as a cell complex K. Pixels are 0-cells of the complex which also have cells of dimension 1 and 2 to explicitly define the topology of the image.

3 Discrete Morse Theory

The discrete Morse theory, formulated by Forman [21], is an adaptation to discrete structures of the Morse theory [37], which relates the critical points of a function f to the topology of the domain.

A **discrete Morse function** on a cell complex K is a function $f\colon K \to \mathbb{R}$ such that, for every $\sigma \in K$, f takes a value less than or equal to $f(\sigma)$ in at most one coface of σ and takes a value greater than or equal to $f(\sigma)$ in at most one face of σ. Figure 2 shows an example of a discrete Morse function.

14	14	3	16	16	23	23	37	37	38	25	25	18
15	15	3	17	17	30	30	42	42	43	35	35	20
8	8	1	6	6	29	29	41	41	43	34	34	20
9	9	1	7	7	31	31	47	47	48	36	36	22
4	4	0	5	5	27	27	44	44	46	32	32	21
13	13	2	11	11	28	28	45	45	46	33	33	21
12	12	2	10	10	24	24	39	39	40	26	26	19

Figure 2 Example of a discrete Morse function. The values of the function are placed over each cell.

Given a discrete Morse function f, a discrete vector field V is a collection of pairs of cells (α^p, β^{p+1}) in K defined whenever $\alpha^p \prec \beta^{p+1}$ and $f(\beta^{p+1}) \leq f(\alpha^p)$. A pair (α^p, β^{p+1}) can be thought of as a discrete tangent vector leaving α and oriented towards β. Pictorially, the vector is represented by an arrow from α to β. Figure 3 shows the vector field of the Morse function presented in Figure 2.

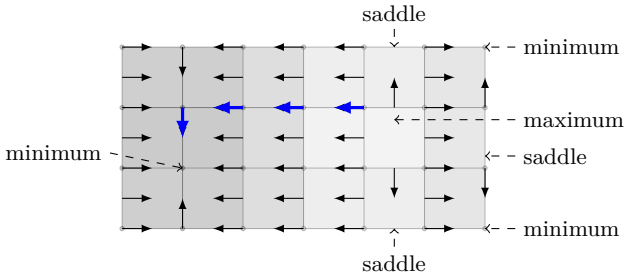


Figure 3 Example of a discrete vector field and critical cells of a discrete Morse function. A V-path is shown in blue.

A cell $\sigma^p \in K$ is **critical of index p** if all of its cofaces take strictly greater values in f and all of its faces are strictly lower in f [21, 22, 44]. In a two-dimensional cell complex, minima are 0-cells, saddles are 1-cells and maxima are 2-cells. Figure 3 shows examples of critical cells of a discrete Morse function. Every cell belongs to a pair in the vector field, except for critical cells.

Another important concept related to vector fields is that of flow. The analogue of a streamline in a continuous function is called a *V-path*, which is a sequence of cells $\alpha_1^p, \alpha_1^{p+1}, \alpha_2^p, \alpha_2^{p+1}, \alpha_3^p, \ldots, \alpha_{r-1}^{p+1}, \alpha_r^p$, such that $(\alpha_i^p, \alpha_i^{p+1}) \in V$, $\alpha_i^{p+1} \succ \alpha_{i+1}^p$, and $\alpha_i^p \neq \alpha_{i+1}^p$, for all $i = 1, \ldots, r-1$. A V-path is a non-trivial closed V-path if $\alpha_r^p = \alpha_1^p$ for $r \geq 2$. Forman [21] presented a discrete vector field V without non-trivial closed paths as the discrete analogue of the continuous gradient vector field. Notice that V-paths alternate between cells of dimension p and $p+1$, therefore, we could explicitly refer to a type of path as a $(p, p+1)$-path, for instance, a $(0,1)$-path alternates between cells of dimension 0 and 1. Figure 3 shows a V-path in blue. The vector field V of the input complex K is computed using the algorithm presented by Robins et al. [44], which returns the pairs of cells in V and the critical cells.

4 Modeling the Discrete Morse Complex

We process two complexes. The first one is the complex K of the input image with its discrete vector field V derived from a discrete Morse function. Pairs of cells can be stored as an attribute of cells in K. The second complex is the discrete Morse complex M that we will compute. Both complexes are modeled through an edge-based data structure, such as the half-edge or the quad-edge.

In order to avoid confusion when referring to cells in K or cells in M, from now on, we will use Greek letters for cells in K and Latin letters for cells in M. Particularly, we will use v^p or u^p to denote a 0-cell in M and e to denote a 1-cell in M. A cell v^p will correspond to a p-cell in K and e will denote a boundary relation between cells in K. Figure 4 shows two complexes K and M and the corresponding cells. The 0-cells in M can be regular or critical according to the cells they correspond to in K. The 1-cells in M explicitly model the boundary relation between two cells in K. The symbols $\circledcirc$, $\oplus$ and $\odot$ will henceforth denote critical 0-cells in M related to 0-, 1- and 2-cells, which are also critical in K. If not critical, the 0-cell will be drawn with the symbol $\circ$.

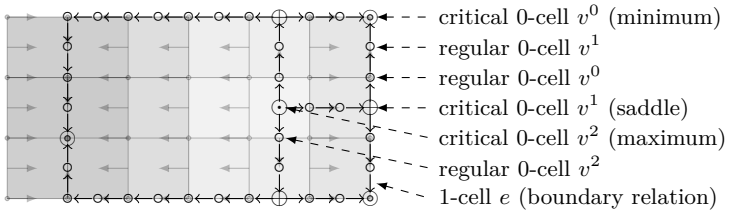


Figure 4 Complex K with M placed over it. The index p of a cell v^p in M is related to the dimension of the corresponding cell in K.

Before presenting the algorithms, we introduce alternative definitions of V-paths to reflect the explicitly modeled paths used in the algorithms. In two dimensions, a discrete Morse complex can have two types of V-paths: $(0,1)$-paths and $(1,2)$-paths. We denote such paths as QV-paths in the complex M we are computing. Definitions for both cases are presented next.

▶ **Definition 1.** *Given a* $(0,1)$*-path* $\alpha_1^0, \alpha_1^1, \alpha_2^0, \ldots, \alpha_{r-1}^1, \alpha_r^0$ *in* V*, a* $(0,1)$*-path in* M*, leaving* v_0^1*, is a sequence of* 0*-cells (denoted by* $v_i^{\{0,1\}}$*) and* 1*-cells (denoted by arrows)* $v_0^1 \to v_1^0 \to v_1^1 \to v_2^0 \to \ldots \to v_{r-1}^1 \to v_r^0$ *where* $v_i^0 \to v_i^1$ *are the cells related to a pair* $(\alpha_i^0, \alpha_i^1) \in V$ *and* v_0^1 *is related to a critical cell* $\alpha^1 \succ \alpha_1^0$*,* $\alpha^1 \neq \alpha_1^1$*.*

Notice from this definition that the $(0,1)$-path leaves a 1-cell and arrives to a 0-cell, making explicit the connection of critical endpoints and following the intuitive notion of a flow going downwards from a saddle to a minimum cell. In a similar way, a $(1,2)$-path definition is presented, but now the path leaves a 2-cell towards a 1-cell.

▶ **Definition 2.** *Given a* $(1,2)$*-path* $\alpha_1^1, \alpha_1^2, \alpha_2^1, \ldots, \alpha_{r-1}^2, \alpha_r^1$ *in* V*, a* $(1,2)$*-path in* M*, arriving to* v_r^1*, is a sequence of* 0*-cells (denoted by* $v_i^{\{1,2\}}$*) and* 1*-cells (denoted by arrows)* $v_0^2 \to v_1^1 \to v_1^2 \to v_2^1 \to \ldots \to v_{r-1}^2 \to v_r^1$ *where* $v_i^1 \to v_i^2$ *are the cells related to a pair* $(\alpha_i^1, \alpha_i^2) \in V$ *and* v_0^2 *is related to the cell* $\alpha^2 \succ \alpha_1^1$*,* $\alpha^2 \neq \alpha_1^2$*.*

5 Extraction of QV-Paths

The algorithm for extracting the QV-paths connecting critical cells is based on searching the paths out of 1-cells. This approach enables efficient computation of the paths in the 2D setting when compared to a breadth-first search [44], since the algorithms consider only paths that may connect critical cells and avoid multiple traversals of cells along sub-paths common to more than one path.

5.1 Initializing QV-Paths from a Saddle

The initialization of QV-paths out of a saddle in K is performed by identifying all of its possible V-paths and including them in the resulting complex M. Up to four V-paths can be expected to go out of a 1-cell since a 1-cell has two faces and at most two cofaces [4]. Given a critical 1-cell α^1 of K, we start all QV-paths arriving and leaving a 0-cell $v_\alpha^1 \in M$ related to α^1 (see Figure 5). In our examples, we use cell complexes of triangles. Regardless, the concepts discussed are not dependent on an image's representation by triangular or quadrilateral 2-cells.

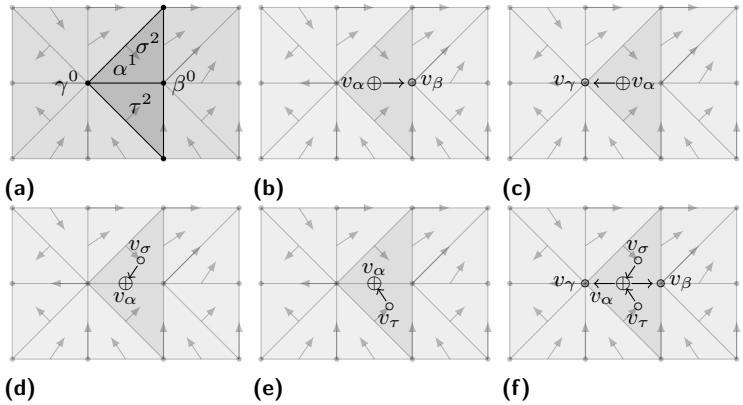


Figure 5 Given a critical 1-cell in V (a): create the QV-paths out of it (b)-(e) and arrange them in the ring of edges out of the vertex related to the 1-cell (f).

The 1-cell α has two faces, β^0 and γ^0; and also two cofaces, σ^2 and τ^2 (Figure 5a). We create the 0-cells in M related to each face and coface of α^1 in K, as well as the four 1-cells that model the boundary relations. This can be observed in Figures 5b to 5e. At the end, the ring of 1-cells out of v_α (a cyclical ordering of the 1-cells leaving a 0-cell v) needs to be adjusted to maintain consistency of the edge-based data structure. All the operations take constant time when using an edge-based data structure. For a better understanding of the operations to create and connect the cells, one could refer to [25]. The connection of cells, for example, can be easily performed through the `Splice` operation described by Guibas and Stolfi [25]. As part of the creation of 0-cells their attributes are properly set, i.e., index p of v^p and indicator of a cell's critical status.

It is important to note that the resulting paths agree with Definitions 1 and 2. Let v_α be v_0^1 in Definition 1, two QV-paths of the form $v_0^1 \to v_1^0$ are now in the complex M, with v_1^0 being v_β in one path and v_γ in another. Both paths are related to a trivial V-path of the form α_1^0 in V. Similarly, we have two paths of the form $v_0^2 \to v_1^1$. However, as explained as part of the discussion of the path extraction algorithm, it is more convenient to think of $(1,2)$-paths as being computed in a backwards fashion. In such a manner, one could think of v_α to be v_r^1 and v_σ and v_τ to be v_{r-1}^2 in Definition 2. Therefore, two paths of the form $v_{r-1}^2 \to v_r^1$ are obtained, both related to a trivial V-path of the form α_r^2 in V.

5.2 Expanding QV-paths

After the initialization, we will iteratively expand the QV-paths. From the previous discussion, we recall $(0,1)$-paths are initialized to grow in a forward manner and $(1,2)$-paths are to be grown backwards. Expanding forward a $(0,1)$-path means to grow the initial QV-path to $v_0^1 \to v_1^0 \to v_1^1 \to v_2^0$ (related to a path $\alpha_1^0, \alpha_1^1, \alpha_2^0$ in V), according to Definition 1. Likewise, expanding backwards the initial $(1,2)$-path means to grow the QV-path to $v_{r-2}^2 \to v_{r-1}^1 \to v_{r-1}^2 \to v_r^1$ (related to a path $\alpha_{r-1}^2, \alpha_{r-1}^1, \alpha_r^2$ in V), according to Definition 2. Therefore, the expansion of the QV-paths is basically obtained by considering a pair (α^0, α^1) or a pair (α^1, α^2), in the input vector field V, as well as a face or coface of α^1 that is not paired with it.

▶ **Definition 3.** *Given a $(0,1)$-path $v_0^1 \to v_1^0 \to \ldots \to v_{k-2}^1 \to v_{k-1}^0$ and its related V-path $\alpha_1^0, \alpha_1^1, \ldots, \alpha_{k-2}^1, \alpha_{k-1}^0$, the expanding triple of the path is the set of cells $\{\alpha_{k-1}^0, \alpha_{k-1}^1, \alpha_k^0\}$ such that $(\alpha_{k-1}^0, \alpha_{k-1}^1) \in V$ and $\alpha_{k-1}^1 \succ \alpha_k^0$, with $\alpha_{k-1}^0 \neq \alpha_k^0$, that allows growing the QV-path to $v_0^1 \to v_1^0 \to \ldots \to v_{k-2}^1 \to v_{k-1}^0 \to v_{k-1}^1 \to v_k^0$ with related V-path $\alpha_1^0, \alpha_1^1, \ldots, \alpha_{k-2}^1, \alpha_{k-1}^0, \alpha_{k-1}^1, \alpha_k^0$.*

Consider the $(0,1)$-path $v_\alpha \to v_\beta$ in Figure 5b. The cells in the expanding triple are shown in Figure 6a following the V-path out of β_{o}^0, namely, β_{o}^0, β^1 and β_{d}^0. The endpoints of an expanding triple are indexed with an "o" or a "d" to make it explicit where the flow enters the path and where it goes to.

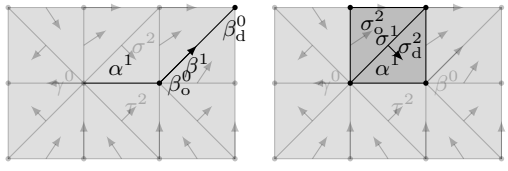


Figure 6 Expanding triples for a $(0,1)$-paths and a $(1,2)$-path.

▶ **Definition 4.** *Given a* $(1,2)$*-path* $v^2_{k+1} \to v^1_{k+2} \to \ldots \to v^2_{r-1} \to v^1_r$*, its related* V*-path* $\alpha^1_{k+2}, \alpha^2_{k+2}, \ldots, \alpha^2_{r-1}, \alpha^1_r$ *and* α^2_{k+1} *related to* v^2_{k+1}*, the expanding triple of the path is the set of cells* $\{\alpha^2_k, \alpha^1_{k+1}, \alpha^2_{k+1}\}$ *such that* $(\alpha^1_{k+1}, \alpha^2_{k+1}) \in V$ *and* $\alpha^2_k \succ \alpha^1_{k+1}$*, with* $\alpha^2_k \neq \alpha^2_{k+1}$*, which allows growing the* QV*-path to* $(0,1)$*-path* $v^2_k \to v^1_{k+1} \to v^2_{k+1} \to v^1_{k+2} \to \ldots \to v^2_{r-1} \to v^1_r$ *related to a* V*-path* $\alpha^1_{k+1}, \alpha^2_{k+1}, \alpha^1_{k+2}, \alpha^2_{k+2}, \ldots, \alpha^2_{r-1}, \alpha^1_r$.

The expanding triple for the $(1,2)$-path $v_\sigma \to v_\alpha$ in Figure 5d is composed of the cells σ^2_{o}, σ^1 and σ^2_{d}, as shown in Figure 6b.

A QV-path may not have an expanding triple, however, if it does, then it is unique.

▶ **Proposition 5.** *The expanding triple of a* $(0,1)$*-path in* M *is unique.*

Proof. Given $v^1_0 \to v^0_1 \to \ldots \to v^1_{k-2} \to v^0_{k-1}$ related to $\alpha^0_1, \alpha^1_1, \ldots, \alpha^1_{k-2}, \alpha^0_{k-1}$ is a $(0,1)$-path that has an expansion, suppose the expanding triple is not unique. Let us then call two of the expansions $\alpha^0_1, \alpha^1_1, \ldots, \alpha^1_{k-2}, \alpha^0_{k-1}, \beta^1_{k-1}, \beta^0_k$ and $\alpha^0_1, \alpha^1_1, \ldots, \alpha^1_{k-2}, \alpha^0_{k-1}, \gamma^1_{k-1}, \gamma^0_k$. If $\beta^1_{k-1} \neq \gamma^1_{k-1}$, it means there are pairs $(\alpha^0_{k-1}, \beta^1_{k-1})$ and $(\alpha^0_{k-1}, \gamma^1_{k-1})$ in V. However, by definition of a vector field, any cell is paired once in V. Consequently, $\alpha^1_{k-1} = \beta^1_{k-1} = \gamma^1_{k-1}$. The paths can still be different if $\beta^0_k \neq \gamma^0_k$. In such a case, the 1-cell α^1_{k-1} should have three faces: α^0_{k-1}, β^0_k and γ^0_k. That is not possible by definition. ◀

▶ **Proposition 6.** *The expanding triple of a* $(1,2)$*-path in* M *is unique.*

Proof. Given $v^2_{k+1} \to v^1_{k+2} \to \ldots \to v^2_{r-1} \to v^1_r$ related to $\alpha^1_{k+2}, \alpha^2_{k+2}, \ldots, \alpha^2_{r-1}, \alpha^1_r$ is a $(1,2)$-path that has an expansion, suppose the expanding triple is not unique. Let us call two of these expansions $\beta^1_{k+1}, \beta^2_{k+1}, \alpha^1_{k+2}, \alpha^2_{k+2}, \ldots, \alpha^2_{r-1}, \alpha^1_r$ and $\gamma^1_{k+1}, \gamma^2_{k+1}, \alpha^1_{k+2}, \alpha^2_{k+2}, \ldots, \alpha^2_{r-1}, \alpha^1_r$. If $\beta^2_{k+1} \neq \gamma^2_{k+1}$ then the 1-cell α^1_{k+2} should have three cofaces: α^2_{k+2}, β^2_{k+1} and γ^2_{k+1}. That is not possible. Since $\alpha^2_{k+1} = \beta^2_{k+1} = \gamma^2_{k+1}$, the paths can still differ if the pairs $(\beta^1_{k+1}, \beta^2_{k+1})$ and $(\gamma^1_{k+1}, \gamma^2_{k+1})$ are in V and $\beta^1_{k+1} \neq \gamma^1_{k+1}$. Again, by definition of V, any cell is paired once, and $\beta^1_{k+1} = \gamma^1_{k+1}$ since $\beta^2_{k+1} = \gamma^2_{k+1}$. ◀

Given a cell α^0 in the input vector field V, we are able to retrieve the 1-cell paired with α^0 in constant time. The second face of a 1-cell can also be computed in constant time with an edge-based data structure. Therefore, retrieving the expanding triple is an $O(1)$ operation. Retrieving the expanding triple in a $(1,2)$-path is similar and also efficient.

Considering again the paths obtained after the initialization steps, these were of the form $v^1_0 \to v^0_1$ and $v^2_{r-1} \to v^1_r$. Now, given the paths in Figures 5b and 5d, the vertices v_β and v_σ and the expanding triples of Figures 6a and 6b, we are able to form $v^1_0 \to v^0_1 \to v^1_1 \to v^0_2$ and $v^2_{r-2} \to v^1_{r-1} \to v^2_{r-1} \to v^1_r$, as shown in Figures 7a and 7b.

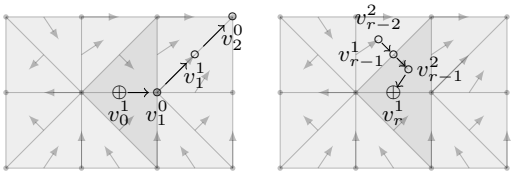


Figure 7 One step expanded QV-paths obtained with the expanding triples of the initial $(0,1)$- and $(1,2)$-paths shown in Figures 5b and 5d.

5.3 Stop Conditions

The paths must be expanded until a termination condition is met. An expanding triple is not found for a $(0,1)$-path when the path reaches a critical cell. A $(1,2)$-path does not have an expanding triple either when it reaches a critical cell or the boundary of the cell complex.

▶ **Proposition 7.** *A $(0,1)$-path $v_0^1 \to v_1^0 \to v_1^1 \to v_2^0 \to \ldots \to v_{k-1}^1 \to v_k^0$ has no expanding triple if the 0-cell, $\alpha^0 \in K$, related to v_k^0 is critical.*

Proof. If α^0 is not critical, then, by definition, it is paired in V. A pair (α_i^0, β_i^1) in a V-path is always followed by a 0-cell α_{i+1}^0 which is a face of β_i^1 other than α_i^0 which must also exist since every 1-cell has two faces. If α^0 is critical, then it is not paired in the vector field and, therefore, no expanding triple can be found. ◀

▶ **Proposition 8.** *A $(1,2)$-path $v_k^2 \to v_{k+1}^1 \to v_{k+1}^2 \to v_{k+2}^1 \to \ldots \to v_{r-1}^2 \to v_r^1$ has no expanding triple if one of two conditions is true:*

1. *the cell α_k^2, related to v_k^2, is critical;*
2. *the cell α_k^2 is paired with a 1-cell, $\alpha^1 \in V$, which has no coface in K other than α_k^2.*

Proof. Condition 1. If α^2 is critical, then it is not paired in V and, therefore, no expanding triple out α^2 can be found. Condition 2. If α^2 is not critical, then it is paired in V, with a 1-cell α^1. The 1-cell is in K since all the faces of a cell must be in K. Given the pair cell α^1, if its two cofaces are in K, then an expanding triple for the path is found, since there is a coface $\alpha_{k-1}^2 \neq \alpha_k^2$. Otherwise, if there is no other coface other than α_k^2, then no expanding triple can be found. ◀

From these conditions, it possible to expand a whole QV-path. However, some additional stop conditions will be introduced. These are based on the fact that V-paths can merge and branch.

Consider a particular 0-cell α^0 in a complex K and suppose it is paired with a 1-cell in the vector field V of the complex. Let also α^0 be a face of k 1-cells in the complex. Except for the 1-cell paired with α^0, each one of the other 1-cells either is paired or is a critical 1-cell. If $l \leq k$ of these 1-cells are paired with a 0-cell, then there are l V-paths containing the cell α^0 and, therefore, at least l paths merge into α^0. Figure 8a shows an example of a cell at which three V-paths merge.

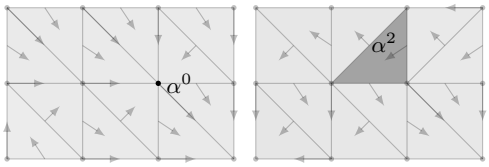


Figure 8 Example of an 0-cell shared by three V-paths and a 2-cell shared by two V-paths.

Now let α^2 be a 2-cell in K paired in V with one of its 1-cell faces. If the α^2 has k 1-cell faces, then these cells may be paired with a 0-cell, paired with a 2-cell or be a critical cell. If $l \leq k$ of them are paired with 2-cells, the V-path containing α^2 will have l V-paths out of it, in other words, the path branches into l V-paths. Figure 8b show a V-path which branches into two V-paths.

Both merging and branching cases are considered stop cases when a QV-path reaches a cell in a path already processed. In Figure 9a, the path ending at the 0-cell v is currently being processed. When the path is expanded from v, it merges with a second path that has been already computed (Figure 9b). From that point on, the expansion of both paths is exactly the same (a consequence of a unique expanding triple). It is also simple to deal with branching cases. The two $(1, 2)$-paths shown in Figure 9c become one in Figure 9d. Notice that, since $(1, 2)$-paths are traversed in a backward fashion, the branching case of $(1, 2)$-paths is algorithmically similar to the merging case of $(0, 1)$-paths and we are able to deal with both cases in a similar way.

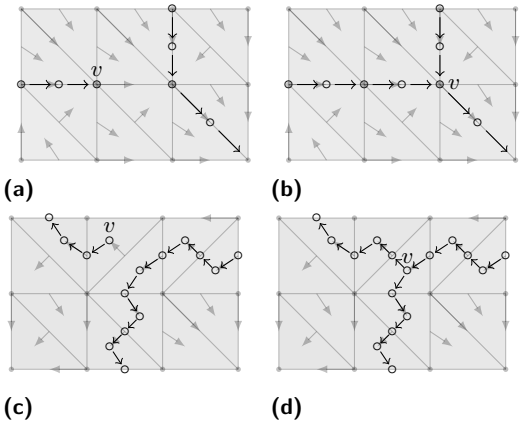


Figure 9 Examples of QV-paths that merge and branch.

A path can be fully extracted by iteratively computing the expanding triples and testing the stop conditions. All these computations can be performed in constant time, therefore, extracting a path takes linear time on the size of the V-path.

The paths of a discrete Morse complex are exactly the QV-paths with critical cells as endpoints. However, there may be $(1, 2)$-paths that do not end at a critical 2-cell. That follows from the second condition of Proposition 8. In order to obtain only the paths of the Morse complex, it suffices removing a $(1, 2)$-path if one of its endpoints is not a critical cell (see Figure 10). This process can be easily performed with the data structure in linear time on the size of the path. Alternatively, one could add a dummy critical cell when the second condition of Property 8 is satisfied.

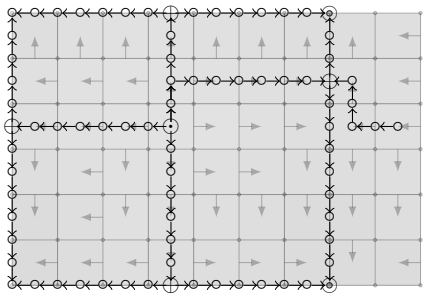


Figure 10 The rightmost path does not end in a critical cell. That could happen in case the path reaches the image boundary.

It is now possible to extract all QV-paths in a discrete Morse complex. It suffices to go through all the saddles of the input complex extracting all of its paths. If the complex has N saddles, then $4N$ paths will be processed. If the size of the i-th path is L_i, then the algorithm will require $O(\sum_{i=1}^{4N} L_i)$ time.

6 Disentangled Complex

Since the extracted paths can merge or branch, we call the complex described in the previous section an *entangled complex*. For some applications, it is important that certain cases, such as merging and branching, do not occur. In the following, we will show how to separate merged and branched paths and keep them independent of each other.

QV-paths merge or branch at a given 0-cell along their extent. We will define that 0-cell $v \in M$ a *knot* if it is not critical and its degree (number of paths arriving and leaving) is greater than two. Figure 9b shows a knot v of degree three. A knot of branching paths is shown in Figure 9d.

If a knot is present, it means that l paths have a common subpath that occurs after a merging case or before a branching case. The idea to disentangle all knots can be thought of as a method that pushes forward (backward) a knot, along the common subpath of merging (branching) paths, until a critical endpoint is reached.

The basic operation works locally at a knot separating the paths one step forward (backward). For such, we need a knot v_1 and the 0-cell v_2 towards which v_1 is pushed. The edge e_1 between v_1 and v_2 is the common subpath to be separated into the k paths entangled in v_1. For the case shown in Figure 11a, two paths merge into v_1, the knot is pushed forward to v_2 along the edge e_1. Using the edge-based data structure, it suffices to disconnect each one of the k paths from the ring of 1-cells of v_1. A new 0-cell v is then created for each path in order to substitute v_1 in that path. A 1-cell e is created to connect v to v_2. In such a manner, all paths are disentangled by one step. Figure 11b shows the result for the example path. A new application of the process will disentangle the paths one step further, resulting in the paths shown in Figure 11c. The operation involves traversing the rings of 1-cells of v_1 and applying constant time computations. The operation takes constant time on average, since the average degree of a planar graph is strictly less than six [28].

It suffices to continue pushing forward (backward) the knot vertices to fully separate paths. Similar to the approach to extracting the QV-paths, we proceed from saddles in M and follow all of its paths disentangling the knots until another critical cell is found. The resulting paths are then disjoint. For visualization purposes and to maintain the planarity of the complexes, the geometrical position of the paths can be perturbed infinitesimally, so that they are positioned parallel to each other. Figure 11d shows the result of disentangling the knots. If the number of saddles is N and each saddle can have up to four paths, the algorithm again will require $O(\sum_{i=1}^{4N} L_i)$ time.

7 Simplified Morse Complex

The complex obtained by the method discussed above can be used for simple visualization, but it is not efficient for many practical tasks, such as computing Betti numbers or hierarchies of the complex. For such tasks, one would intend to obtain the critical cells connected by QV-paths in constant time, without traversing all paths.

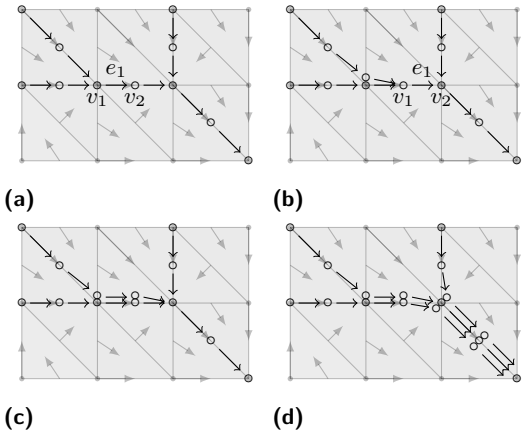


Figure 11 Example of untangling knots of a merged path. In (a), the common subpath between v_1 and v_2 is split so that the two paths arriving at v_1 are then separated in (b). The process is applied once again to obtain the separated paths in (c), and, by iterating along the whole common subpaths, fully separated paths are obtained (d).

We present a simplified version of our discrete Morse complex such that the 1-cells connect critical cells. The path between the critical cells are stored in a list as an attribute of the 1-cells. With such a complex, we obtain the objective of a model that is easy to manipulate through local operations, but that can also be used for visualization purposes.

We consider subpaths called *simplification kernels*, depicted in Figures 12a and 12b. The kernel is formed by the set of cells $\{e, v_1, e_1, v_2, e_2, v_3\}$. We chose such a kernel to maintain the property that a path alternates between cells of different dimensions. If the destination of e is not a critical cell, then we can argue that a kernel can be found. Consider a $(0,1)$-path $u_0^1 \to u_1^0 \to u_1^1 \to u_2^0 \to \ldots \to u_{r-1}^1 \to u_r^0$, as in Definition 1. The saddle in the origin of e is u_0^1 and the destination is u_1^0 in the path. By definition, the subpath $u_1^0 \to u_1^1 \to u_2^0$ should be present in the path. These are exactly the cells needed to form the kernel. The same ideas can be applied to a $(1,2)$-path.

The 1-cell e has as origin a saddle in M. The cells v_1 and v_2, as well as e_1 and e_2, are removed from the complex and the 1-cell e is connected to v_3. The result of the simplification of the complex from Figure 12a is shown in Figure 12b.

Finding kernels and simplifying them must be repeated until the 1-cell e finally connects critical 0-cells. The endpoints of e shown in Figure 12b still are not both critical. Therefore, a new kernel is found and simplified. The result is the edge e shown in Figure 12c that connects critical cells and the process stops. The time complexity is linear in the size of the path. Again, the process must be repeated for each path, implying that a fully simplified complex is obtained in $O(\sum_{i=1}^{4N} L_i)$ time. The resulting complex for our example is shown in Figure 12d.

8 Persistence and Topological Hierarchies

We present applications of the discrete Morse complex for analysis of image functions and important topological computations, such as persistence of critical cells [13, 23, 38, 39] and simplification of a topological structure to produce a hierarchical representation [35, 46, 49].

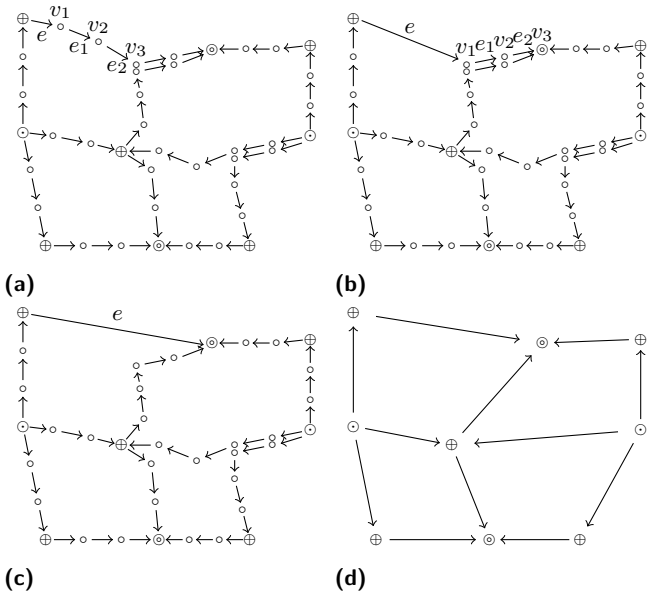


Figure 12 Simplification of subpaths and resulting discrete Morse complex. In (a) a kernel is used to exchange a subpath by a unique 1-cell (b). The process is repeated in (b) and stops since critical cells are now directly connected (c). In (d) the discrete Morse complex connects only critical cells.

We use elevation functions comprising one synthetic image and four real-world terrain height field data [41, 48]. The synthetic image was computed as $h(x, y) = \sin x + \sin y$ multiplied by an exponential $g(x, y) = \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$, with x and y in the interval $[-40, 40]$ and $\sigma = 6$. The image is shown in Figure 13a and its resulting discrete Morse complex in presented in Figure 13b. The information on the image set and the number of critical cells of each type in the resulting discrete Morse complexes are given in Table 1.

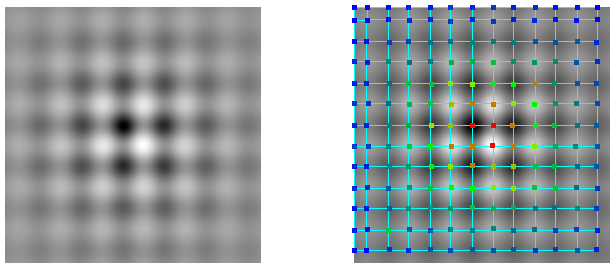


Figure 13 Discrete Morse complex (b) of the synthetic sinusoidal image (a). The colors in (b) represent the persistence of critical points, ranging from low persistence (blue) to high persistence (red), with intermediate values in yellow and green.

The complexes were employed to compute the persistence of critical points and hierarchical decompositions of the complexes through cancellation of low persistent topological features. These topological features are 0-cycles and 1-cycles, which, in a two-dimensional space, are related to connected components and holes in the level sets of the images. The number of cycles in the level sets, along with persistence, was computed with the algorithm described

Table 1 Images used in the experiments. The second column lists the resolution (in pixels) of the images, third to fifth columns report the number of cells in each dimension in the respective cell complexes, whereas sixth to eighth columns report the number of critical points in the Morse complex.

Image	Resolution (pixels)	Image Complex			Morse Complex		
		0-cells	1-cells	2-cells	Minima	Saddle	Maxima
Sine	256 × 256	65536	130560	65025	49	84	36
Crater Lake	336 × 459	154224	307653	153430	355	713	359
Cumberland	1201 × 1201	1442401	2882400	1440000	8058	23138	15081
Death Valley	1201 × 1201	1442401	2882400	1440000	27964	46319	18356
Mars	936 × 949	888264	1774643	886380	568	4293	3726

by Zomorodian [54]. The algorithm requires a total ordering of the critical points in the Morse complex, which was obtained by assigning the grayscale value of each critical cell to the corresponding point in the Morse complex. For details on the subject, one should refer to specialized literature [29, 53, 54].

The image in Figure 13b depicts in different colors the persistence of each critical point in the discrete Morse complex of the sinusoidal image (Figure 13a). The color scale varies from blue to red, representing low persistence and high persistence. Intermediate values are shown in yellow and green intensities. The critical points of higher persistence are positioned in the center of the image and the persistence decays from these points to points in the boundaries of the image. This effect is due to the exponential function added to the sinusoidal function. The 0-cycles and 1-cycles of the image are summarized in the graphs of Figure 14. The cycles are sorted by persistence and it can be noticed that the persistence increases following an exponential behavior, as expected.

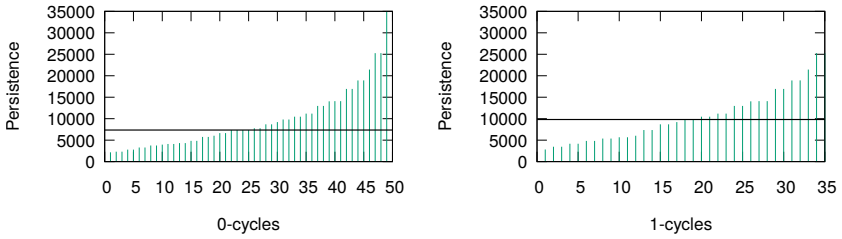
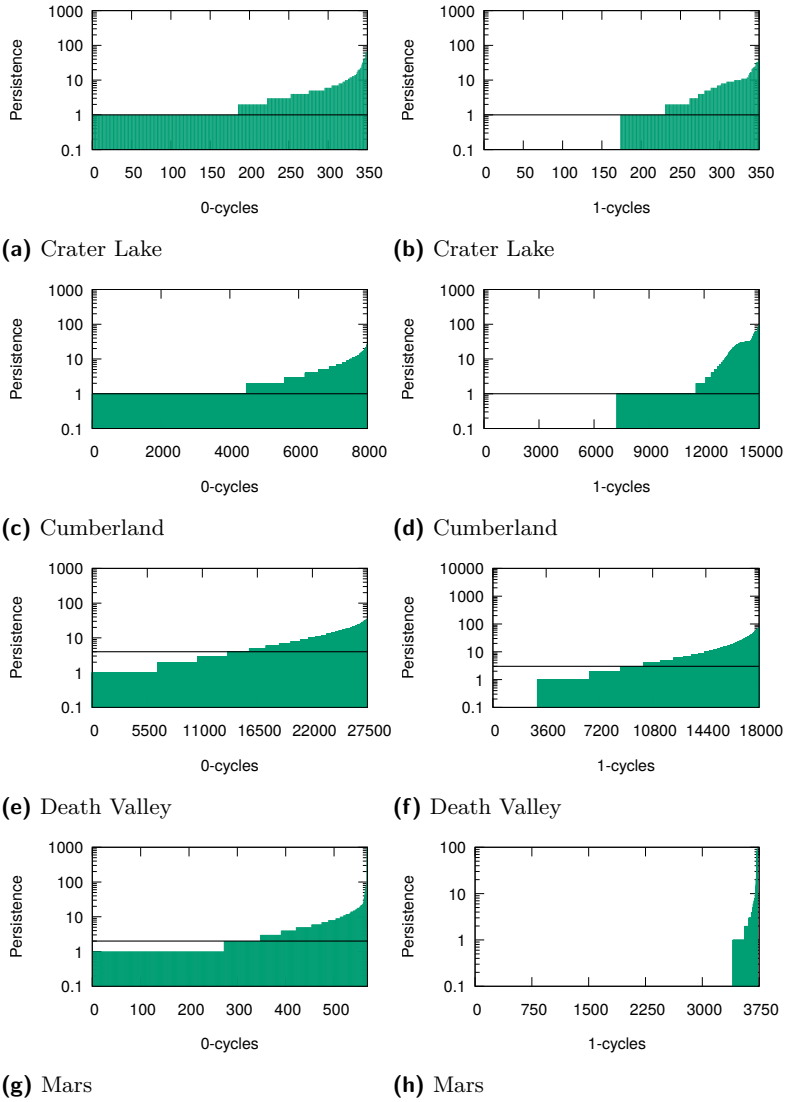


Figure 14 Graphs that illustrate the respective persistence for each 0- and 1-cycle in the sinusoidal image. The sorted persistence of the cycles grows exponentially, except for the last 0-cycle, which has infinite persistence. The horizontal line presents the median for reference.

The graphs for number of cycles versus persistence for the real images are shown in Figure 15. The persistence axis is shown in logarithmic scale, so that the values can be better analyzed. These graphs show a common characteristic: a small amount of cycles with high persistence and a great amount of cycles with very low persistence. These very low persistent cycles may be due to topological noise of the data. An important task is then the removal of such topological noise, as explored in [9] and [18]. Cleaning the topological noise may be interesting for better understanding the function or the phenomenon being studied. The data structure of our complex allows easy and fast manipulation of the complex for simplifications. We implemented the operation as described in [18].



(a) Crater Lake **(b)** Crater Lake

(c) Cumberland **(d)** Cumberland

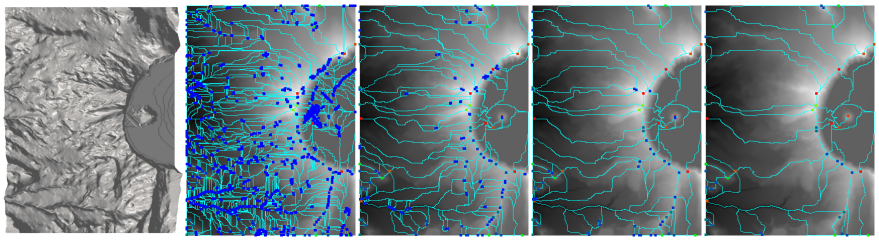
(e) Death Valley **(f)** Death Valley

(g) Mars **(h)** Mars

Figure 15 Graphs that illustrate the respective persistence for each 0- and 1-cycle of real elevation terrain images. The sorted persistence shows a small amount of cycles with high persistence and a great amount of cycles with very low persistence. The horizontal line presents the median for reference.

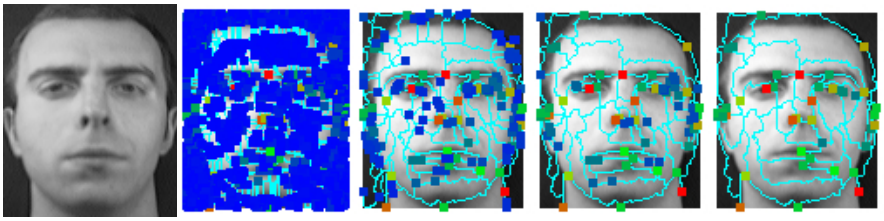
It can be noticed that noisy topological features are mostly concentrated in low persistent points. By suppressing low persistent features, the discrete Morse complex becomes cleaner. This effect is reflected in the hierarchy of simplified discrete Morse complexes, shown in Figure 16. In the initial complexes, the island in the middle of the lake does not stand out, since there are many low persistence features. As the simplification progresses, topological features of low prominence are removed and important topological features are preserved.

We have also computed the hierarchy for a face image, as shown in Figure 17. Initially, there is a high concentration of feature areas with non-relevant details. The details are removed as the levels in hierarchy are traversed and features located at the eyes, nose and mouth stand out. This example shows how the simplification of the complexes can be used to capture significant topological information of images and suggests that it can be useful for image classification. The image was obtained from the AT&T Laboratories Cambridge [2].



(a) DEM **(b)** original **(c)** pers. > 8 **(d)** pers. > 16 **(e)** pers. > 32

Figure 16 Hierarchy for the Crater Lake image.



(a) face **(b)** original **(c)** pers. > 8 **(d)** pers. > 16 **(e)** pers. > 32

Figure 17 A hierarchy of discrete Morse complexes computed for a face image.

Finally, we discuss computational time and memory usage (Table 2) for each step in the construction of the discrete Morse complex. The algorithms were implemented[1] using the C programming language and the experiments were performed on a computer with an Intel i7 core (1.8GHz and 8GB RAM memory) with Ubuntu 18.04. The quad-edge [25] data structure was used for the complexes. In addition, for the images already covered in the previous discussion, we provide information for the Berkeley Segmentation Data Set and Benchmarks 500 (BSDS500) [1]. The dataset consists of 500 images of two different sizes, i.e., 152 images of 312×481 pixels and 348 images of 481×312 pixels.

Table 2 shows the mean computational time and memory usage for the BSD dataset. Even though our implementation used the quad-edge data structure for the image complex and for the discrete vector field, it would be possible to reduce memory cost with memory-efficient data structures [20]. In addition, the image complex and the vector field are no longer necessary after computing the discrete Morse complex. The Topology Toolkit (TTK) [7, 47] supports Morse-Smale computation. In Table 3, we provide execution time and memory usage using TTK (version 0.9.8) filters in the ParaView software (version 5.6.1) [3]. The software output values are reported. Our implementation of our algorithms is efficient, as can be seen in columns 4 to 6 in Table 2. The computation of the vector field, the input to our algorithms, needs to be improved, as has not yet been optimized for images of large dimensions. This efficiency improvement is possible, since TTK also computes the vector field at the reported times. TTK triangulates complexes as well, instead of using a quadrilateral grid, which could potentially increase computational time. It is our plan to improve efficiency related to path computations and fine-tune our implementation for better run time performance.

[1] Code available at `https://pessoal.dainf.ct.utfpr.edu.br/rdsilva/codes/morse.zip`.

Table 2 Computational time and memory usage for each step of the algorithm. Time and memory is reported for each step: image cell complex (Img), computation of the vector field (VF), computation of the paths (Paths), disentangling the paths (Dis.) and contracting the paths (Con.). The contract step does not increase the memory, therefore we report the complex storage size after disentangling and contracting steps as a unique value.

	Time (seconds)					Memory (megabytes)			
Image	Img	VF	Paths	Dis.	Con.	Img	VF	Paths	Dis./Con.
BSD	0.067	1.542	0.500	0.427	0.377	92.90	169.49	119.04	173.14
Sine	0.027	0.569	0.114	0.004	0.004	39.30	70.97	4.98	4.98
Crater Lake	0.067	1.493	0.330	0.089	0.064	92.88	168.80	26.13	44.43
Cumberland	0.611	15.743	3.737	2.117	1.711	868.94	1571.61	428.79	1091.62
Death Valley	0.618	15.948	4.349	2.030	1.786	868.93	1572.64	609.087	998.90
Mars	0.383	9.236	2.075	0.682	0.514	534.81	967.08	148.58	345.81

Table 3 Computational time (four threads) and memory usage from TTK using ParaView.

Image	Time (seconds)	Memory (megabytes)
Sine	0.158	1.48
Crater Lake	0.458	6.38
Cumberland	4.841	148.10
Death Valley	5.479	132.16
Mars	2.733	-

9 Conclusions

We have presented and analyzed algorithms for computing the discrete Morse complex for two-dimensional images. We model a Morse complex with an efficient edge-based data structure, e.g., a winged-edge or a quad-edge structure, to guarantee constant-time local operations applied to a Morse complex.

The presented algorithms ensure that only paths in a vector field that may lead to paths in a Morse complex are processed. The algorithms deal with merging and branching in order to produce a consistent representation of the model. The simplified Morse complex can be manipulated in a straightforward manner and is useful to perform local topological operations. The model used for the complex is suitable for data analysis and visualization.

The presented theoretical and experimental results show the effectiveness of our algorithms. For example, the complex is suitable for the computation of homological persistence numbers [53], removal of topological noise [51], and hierarchical representation of the discrete Morse complex [18]. Computation times show that our algorithm has a performance comparable to that used in the Topological Toolkit (TTK). However, memory requirements of our algorithms should be reduced, and we plan to address this issue in future work.

References

1 Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 33(5):898–916, May 2011.

2 ATTLC. The Database of Faces, 2019. URL: `http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html`.

3 Utkarsh Ayachit. *The ParaView Guide: A Parallel Visualization Application*. Kitware, Inc., Clifton Park, NY, USA, 2015.

4 Tathagata Basak. Combinatorial Cell Complexes and Poincaré Duality. *Geometriae Dedicata*, 147(1):357–387, 2010.

5 Bruce G. Baumgart. A Polyhedron Representation for Computer Vision. In *National Computer Conference and Exposition*, pages 589–596, Anaheim, CA, USA, 1975. ACM.

6 K. Beketayev, G.H. Weber, M. Haranczyk, P.-T. Bremer, M. Hlawitschka, and B. Hamann. Topology-based Visualization of Transformation Pathways in Complex Chemical Systems. *Computer Graphics Forum*, 30(3):663–672, 2011.

7 Talha Bin Masood, Joseph Budin, Martin Falk, Guillaume Favelier, Christoph Garth, Charles Gueunet, Pierre Guillou, Lutz Hofmann, Petar Hristov, Adhitya Kamakshidasan, Christopher Kappe, Pavol Klacansky, Patrick Laurin, Joshua Levine, Jonas Lukasczyk, Daisuke Sakurai, Maxime Soler, Peter Steneteg, Julien Tierny, Will Usher, Jules Vidal, and Michal Wozniak. An Overview of the Topology ToolKit. In *TopoInVis*, 2019.

8 P. T. Bremer, E. M. Bringa, M. A. Duchaineau, A. G. Gyulassy, D. Laney, A. Mascarenhas, and V. Pascucci. Topological Feature Extraction and Tracking. *Journal of Physics: Conference Series*, 78(1):1–5, 2007.

9 P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A Multi-Resolution Data Structure for Two-Dimensional Morse-Smale Functions. In *IEEE Visualization*, pages 139–146, Washington, DC, USA, 2003.

10 Gunnar Carlsson, Gurjeet Singh, and Afra Zomorodian. Computing Multidimensional Persistence. In Yingfei Dong, Ding-Zhu Du, and Oscar Ibarra, editors, *Algorithms and Computation*, volume 5878 of *Lecture Notes in Computer Science*, pages 730–739. Springer Berlin Heidelberg, 2009.

11 Gunnar Carlsson, Afra Zomorodian, Anne Collins, and Leonidas Guibas. Persistence Barcodes for Shapes. In *Eurographics Symposium on Geometry Processing*, pages 124–135, New York, NY, USA, 2004.

12 Chao Chen and Michael Kerber. Persistent Homology Computation with a Twist. In *27th European Workshop on Computational Geometry*, pages 197–200, Morschach, Switzerland, 2011.

13 Moo K. Chung, Peter Bubenik, and Peter T. Kim. Persistence Diagrams of Cortical Surface Data. In *Conference on Information Processing in Medical Imaging*, pages 386–397, Williamsburg, VA, USA, 2009.

14 David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of Persistence Diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007.

15 Anne Collins, Afra Zomorodian, Gunnar Carlsson, and Leonidas J. Guibas. A Barcode Shape Descriptor for Curve Point Cloud Data. *Computers and Graphics*, 28(6):881–894, 2004.

16 Leila De Floriani and Annie Hui. Data Structures for Simplicial Complexes: An Analysis and a Comparison. In *Third Eurographics Symposium on Geometry Processing*, Vienna, Austria, 2005. Eurographics Association.

17 O. Delgado-Friedrichs, V. Robins, and A. Sheppard. Skeletonization and partitioning of digital images using discrete morse theory. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):654–666, 2015.

18 Herbert Edelsbrunner, John Harer, and Afra Zomorodian. Hierarchical Morse Complexes for Piecewise Linear 2-Manifolds. In *Symposium on Computational Geometry*, pages 70–79, Medford, MA, USA, 2001.

19 Barbara Di Fabio and Claudia Landi. Persistent Homology and Partial Similarity of Shapes. *Pattern Recognition Letters*, 33(11):1445–1450, 2012.

20 Riccardo Fellegara, Leila De Floriani, and Kenneth Weiss. Efficient Computation and Simplification of Discrete Morse Decompositions on Triangulated Terrains. In *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 223–232, Dallas, TX, USA, 2014. ACM.

21 R. Forman. Morse Theory for Cell Complexes. *Advances in Mathematics*, 134(1):90–145, 1998.

22 R. Forman. Morse Theory and Evasiveness. *Combinatorica*, 20(4):489–504, 2000.

23 Jennifer Gamble and Giseon Heo. Exploring Uses of Persistent Homology for Statistical Analysis of Landmark-Based Shape Data. *Journal of Multivariate Analysis*, 101(9):2184–2199, 2010.

24 Leo Grady and Jonathan R. Polimeni. *Discrete Calculus - Applied Analysis on Graphs for Computational Science*. Springer, 2010.

25 Leonidas Guibas and Jorge Stolfi. Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams. *ACM Transactions on Graphics*, 4(2):74–123, 1985.

26 David Günther, Jan Reininghaus, Hubert Wagner, and Ingrid Hotz. Efficient Computation of 3D Morse–Smale Complexes and Persistent Homology using Discrete Morse Theory. *The Visual Computer*, 28(10):959–969, October 2012.

27 Attila Gabor Gyulassy. *Combinatorial Construction of Morse-smale Complexes for Data Analysis and Visualization*. PhD thesis, University of California, Davis, CA, USA, 2008.

28 Frank Harary. *Graph Theory*. Addison-Wesley, 1969.

29 Allen Hatcher. *Algebraic Topology*. Cambridge Univ. Press, Cambridge, 2000.

30 Henry King, Kevin Knudson, and Neža Mramor Kosta. Birth and Death in Discrete Morse Theory. *Journal of Symbolic Computation*, 78:41–60, 2017.

31 Henry King, Kevin Knudson, and Neža Mramor. Generating Discrete Morse Functions from Point Data. *Experimental Mathematics*, 14(4):435–444, 2005.

32 V. A. Kovalevsky. Finite Topology as Applied to Image Analysis. *Computer Vision, Graphics and Image Processing*, 46(2):141–161, 1989.

33 David Letscher and Jason Fritts. Image Segmentation Using Topological Persistence. In *Conference on Computer Analysis of Images and Patterns*, pages 587–595, Vienna, Austria, 2007.

34 Thomas Lewiner, Helio Lopes, and Geovan Tavares. Applications of Forman's Discrete Morse Theory to Topology Visualization and Mesh Compression. *IEEE Transactions on Visualization and Computer Graphics*, 10(5):499–508, 2004.

35 P. Magillo, E. Danovaro, L. Floriani, L. Papaleo, and M. Vitali. A Discrete Approach to Compute Terrain Morphology. In J. Braz, A. Ranchordas, H. Araújo, and J.M. Pereira, editors, *Computer Vision and Computer Graphics. Theory and Applications*, volume 21 of *Communications in Computer and Information Science*, pages 13–26. Springer Berlin Heidelberg, 2009.

36 Martti Mäntylä. *An Introduction to Solid Modeling*. Computer Science Press, Inc., New York, NY, USA, 1987.

37 J. W. Milnor. *Morse Theory*. Annals of Mathematics Studies. Princeton University Press, 1963.

38 Elías Gabriel Minian. Some Remarks on Morse Theory for Posets, Homological Morse Theory and Finite Manifolds. *Topology and its Applications*, 159(12):2860–2869, 2012.

39 Konstantin Mischaikow and Vidit Nanda. Morse Theory for Filtrations and Efficient Computation of Persistent Homology. *Discrete & Computational Geometry*, 50(2):330–353, 2013.

40 Helena Molina-Abril and Pedro Real. Homological Optimality in Discrete Morse Theory Through Chain Homotopies. *Pattern Recognition Letters*, 33(11):1501–1506, 2012.

41 NASA. Jet Propulsion Laboratory, 2019. URL: `http://www.jpl.nasa.gov`.

42 Sylvain Paris and Fredo Durand. A Topological Approach to Hierarchical Segmentation using Mean Shift. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Los Alamitos, CA, USA, 2007.

43 J. Reininghaus, N. Kotava, D. Gunther, J. Kasten, H. Hagen, and I. Hotz. A Scale Space Based Persistence Measure for Critical Points in 2D Scalar Fields. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2045–2052, 2011.

44 Vanessa Robins, Peter John Wood, and Adrian P. Sheppard. Theory and Algorithms for Constructing Discrete Morse Complexes from Grayscale Digital Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1646–1658, 2011.

45 Nithin Shivashankar and Vijay Natarajan. Parallel Computation of 3D Morse-Smale Complexes. *Computer Graphics Forum*, 31(3pt1):965–974, 2012.

46 A. Sole, V. Caselles, G. Sapiro, and F. Arandiga. Morse Description and Geometric Encoding of Digital Elevation Maps. *IEEE Transactions on Image Processing*, 13(9):1245–1262, 2004.

47 J. Tierny, G. Favelier, J. A. Levine, C. Gueunet, and M. Michaux. The topology toolkit. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):832–842, 2018.

48 USGS. United States Geological Survey, 2019. URL: `http://www.usgs.gov`.

49 Lidija Čomić, Leila De Floriani, and Federico Iuricich. Simplifying Morphological Representations of 2D and 3D Scalar Fields. In *ACM International Conference on Advances in Geographic Information Systems*, pages 437–440, Chicago, IL, USA, 2011.

50 Hubert Wagner, Paweł Dłotko, and Marian Mrozek. Computational Topology in Text Mining. In Massimo Ferri, Patrizio Frosini, Claudia Landi, Andrea Cerri, and Barbara Fabio, editors, *Computational Topology in Image Context*, volume 7309 of *Lecture Notes in Computer Science*, pages 68–78. Springer Berlin Heidelberg, 2012.

51 Bei Wang. *Separating Features from Noise with Persistence and Statistics*. PhD thesis, Duke University, Durham, NC, USA, 2010.

52 Kenneth Weiss, Federico Iuricich, Riccardo Fellegara, and Leila De Floriani. A Primal/Dual Representation for Discrete Morse Complexes on Tetrahedral Meshes. *Computer Graphics Forum*, 32(3pt3):361–370, 2013.

53 A. Zomorodian. *Computing and Comprehending Topology: Persistence and Hierarchical Morse Complexes*. PhD thesis, University of Illinois at Urbana-Champaign, 2001.

54 A. Zomorodian. Computational Topology. In M. Atallah and M. Blanton, editors, *Algorithms and Theory of Computation Handbook*, volume 2, chapter 3. Chapman & Hall/CRC Press, second edition, 2010.

Interpolation of Scientific Image Databases

Eric Georg Kinner
Technische Universität Kaiserslautern, Germany

Jonas Lukasczyk
Arizona State University, Tempe, AZ, US

David Honegger Rogers
Los Alamos Research Laboratory, NM, US

Ross Maciejewski
Arizona State University, Tempe, AZ, US

Christoph Garth
Technische Universität Kaiserslautern, Germany

Abstract

This paper explores how recent convolutional neural network (CNN)-based techniques can be used to interpolate images inside scientific image databases. These databases are frequently used for the interactive visualization of large-scale simulations, where images correspond to samples of the parameter space (e.g., timesteps, isovalues, thresholds, etc.) and the visualization space (e.g., camera locations, clipping planes, etc.). These databases can be browsed post hoc along the sampling axis to emulate real-time interaction with large-scale datasets. However, the resulting databases are limited to their contained images, i.e., the sampling points. In this paper, we explore how efficiently and accurately CNN-based techniques can derive new images by interpolating database elements. We demonstrate on several real-world examples that the size of databases can be further reduced by dropping samples that can be interpolated post hoc with an acceptable error, which we measure qualitatively and quantitatively.

2012 ACM Subject Classification Computing methodologies → Image processing

Keywords and phrases Image Interpolation, Image Database, Cinema Database

Digital Object Identifier 10.4230/OASIcs.iPMVM.2020.19

Supplementary Material *Software*: `https://github.com/EricKinner/InterpScImgDB`
archived at `swh:1:dir:c86bc70a8f4399d4f3f8779b21e4bae2da16791b`

1 Introduction

Today, almost any domain of research and engineering is using scientific simulations. The complexity and scale increased over time due to innovations and continuous improvements in both hardware and algorithmic solutions. With this, the results also grew in size. Now, the management of data is increasingly becoming a major concern for large-scale applications. It has been examined that persistently saving or archiving data produces over-proportionally high costs (Kunkel et al. [13]), which strain institutional resources. Approaching exa-scale computing capabilities, it becomes apparent that storage constraints limit the scope of more and more simulations. Researchers are forced to decrease data output frequencies, shorten the simulation time, or shrink the region and ensemble sizes, which in return hinders the analysis and scientific workflow in general.

Scientific simulations are a valuable tool for researchers of many domains to derive and verify hypothesis, which are otherwise unobservable. Therefore, there were many techniques proposed to mitigate this problem [6][14][7][31][10][12] (further discussed in related work). One of them by Ahrens et al. [2], who proposed the creation of a image databases. These contain in situ rendered visualizations of the simulation for e.g. each timestep. Those images are then directly used to visually analyze the results afterwards. This has several benefits

2nd International Conference of the DFG International Research Training Group 2057 – Physical Modeling for Virtual Manufacturing (iPMVM 2020).
Editors: Christoph Garth, Jan C. Aurich, Barbara Linke, Ralf Müller, Bahram Ravani, Gunther Weber, and Benjamin Kirsch; Article No. 19; pp. 19:1–19:17

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

over the storage of raw simulation data. Adhinarayanan et al. [1] confirmed through a theoretical model the overall lower power consumption, hardware- and storage requirements in comparison to conventional methods.

But the creation of image databases also introduces a new trade-off between the sampling-rate and the required storage space. The denser the simulation is sampled, the more intuitive and detailed is the post hoc analysis. But excessive sampling over a large number of parameters (e.g. isovalues, thresholds, timesteps, camera angles) requires potentially the same or more disk space.

In this paper we explored how image interpolation can resolve or mitigate this trade-off. The reconstruction of intermediate frames through interpolation could be used as compression to reduce the size of these image databases. This allows in return to effectively save more samples. The creation of new frames is also possible, which allows the arbitrary change of the sampling rate. More frames usually mean smoother transitions and improved user experience. In general, image interpolation could be used for compression and data augmentation on these specific or any other image databases. It may be employed as a compression or pre/post processing step for an analysis or visualization.

However, we are not aware of any image interpolation technique that was developed for scientific visualizations. Also, to our knowledge, general purpose image interpolation has not yet been applied on scientific image datasets. Common images and scientific visualizations differ substantially in the depicted objects and the way they were rendered. Hence, we examined the feasibility and properties along minor improvements in the following. In particular we present the following contributions:

- Feasibility of image interpolation on scientific image databases
- Comparison between state-of-the-art approaches on scientific visualizations
- Publicly available docker images for reproducible results
- Case study on background substitution for algorithmic approaches
- Case study on fine-tuning CNNs for scientific visualizations

2 Related Work

The increasing demand of disk space for large-scale simulations has naturally gained the attention of many researchers. Several mitigation techniques have been developed to reduce the memory consumption of high-performance systems. Burtscher et al. [6] presented a floating point compressor, which can also be used for inter-process communication. Sequences of floating point numbers are packed and compressed together to reduce size. Lossy techniques, like the SZ compression by Di et al. [7], provide even greater data reduction. Sophisticated approaches like the "In situ Sort-And-B-spline Error-bounded Lossy Abatement" (ISABELA), developed by Lakshminarasimhan et al. [14], or the work of Tao et al. [31] reveal correlations of seemingly random data and provide very high compression for general data assemblies. Approaches that are specialized on a certain domain can often exploit similarities in the data more and therefore reach better results. For example, Jeruzalski et al. [10] introduced a novel compression for contact-dominated rigid body simulations. Since it can operate on a subset of the data, the resulting memory overhead is minimal. Kumar et al. [12] on the other hand proposed a lossy compression, that utilizes discrete cosine transform and principal component analysis on molecular dynamics simulation data.

This work is motivated by the approach of Ahrens et al. [2], who rendered the simulation data in situ and created image databases. Instead of storing raw data (e.g. position/velocity of every particle in a simulation), everything is captured through images, which are sampled along the parameter space of interest (e.g. timesteps, camera locations, isovalues, thresholds,

etc.). Adhinarayanan et al. [1] confirmed the lower storage and power requirements of in situ rendered simulation analysis in comparison to other methods. However, the denser the simulation is sampled, the more precise is the analysis. This imposes a strong trade-off between accuracy and the required storage space. Storing more images eventually becomes inefficient.

Our approach, to use general purpose image interpolation on these scientific image databases, mitigates the described predicament. The most similar techniques to our method include the work of Lukasczyk et al. [20]. They proposed a view approximation approach to reduce the spacial sampling rate. Through depth images, the scene is reconstructed. This enables the free exploration of the data with arbitrary viewpoint. But the generation of depth images is infeasible for e.g. volumetric datasets, which limits the applicability of this approach. Methods specialized on volumes, like from Fernandes et al. [8], enable post hoc camera changes through i.e. space-time-coherent volumetric depth images. But this solution is limited to volumes and can only be applied in situ. The compression of image databases through video codecs was evaluated by Berres et al. [5], which resulted in great data reduction rates. The video compression of image databases works also post hoc, but does not improve on user experience. Our image interpolation approach enables post hoc changes in frame rate, which augments the data and increases the expressiveness of data. Similar techniques are used on medical data from CT or MRT scans to improve the hardware restricted resolution of scans. A specialized technique for this was developed by Leng et al. [15], which focuses on the transformations of these scans.

The application of image interpolation on scientific datasets can compress its size and improve user experience. It is also applicable regardless of the underlying data type. It can be applied in situ as well as on existing datasets. Also, advancements in interpolation techniques can directly be incorporated due to the high modularity.

Image interpolation is an ongoing challenge in image processing and finds its application commonly in slow motion generation or the resampling of movies. The active research community continuously publishes ever improving solutions to this problem. Inspirational papers include the work of Mahajan et al. [23], which presented a algorithmic path-based technique and most recently the RRIN model of Li et al. [16].

Among numerous other techniques the following image interpolation techniques were chosen to test their applicability on scientific visualizations. Their source code is publicly accessible, which allows our results to be reproduced and extended.

- **$\text{Inter}_{\text{sepconv}}$: Video Frame Interpolation via Adaptive Separable Convolution** by Niklaus et al. (2017) [27] is a convoluational neural network for image interpolation. They use separable kernels to reduce the memory complexity to $O(2n)$. With this, larger motions can be detected. The approach is set to interpolate a single frame in the middle between two reference frames. Benchmarks show a performance of 0.9 seconds for a 1920x1080 frame on a Nvidia Titan X (Pascal). In this paper, we will refer to this approach by "$\text{Inter}_{\text{sepconv}}$" for convenience.
- **$\text{Inter}_{\text{phase}}$: Phase-based frame interpolation for video** of Meyer et al. (2015)[25] solves the interpolation of images algorithmically. The core idea is to interpret the color of images as functions and motion as the phase shift of these functions. Interpolation of the images is then solved by interpolating the phase. Without global optimization, peak performance of 1 seconds for a image of 720p can be reached on a Nvidia GeForce GTX 770. The approach allows for arbitrarily many equally spaced interpolated frames between two reference frames. In this paper, we will refer to this approach by "$\text{Inter}_{\text{phase}}$" for convenience.

- **Inter$_{\text{voxel}}$: Video Frame Synthesis using Deep Voxel Flow** is the initial work of Liu et al. (2017) [19] and involves a convolutional neural network for image interpolation and extrapolation. It is implicitly trained end-to-end on voxel flow by images directly. Although theoretically arbitrary interpolation is possible, the open source implementation only supports one intermediate frame. The authors did not make statements about the performance of the model aside from quality evaluations. In this paper, we will refer to this approach by "Inter$_{\text{voxel}}$" for convenience.
- **Inter$_{\text{cyclic}}$: Deep Video Frame Interpolation using Cyclic Frame Generation** is the later work of Liu et al. (2019) [18], which introduced a forward-backward consistency as a mean for regularization. The model is by design restricted to a single interpolation between two reference frames. The authors did not make any statements about the performance of the model. In this paper, we will refer to this approach by "Inter$_{\text{cyclic}}$" for convenience.
- **Inter$_{\text{dain}}$: Depth-Aware Video Frame Interpolation** by Boa et al. (2019) [4] is a composed image interpolation approach. It uses four existing models to estimate depth, optical flow, context, and the kernels. The approach allows for arbitrarily many equally spaced interpolated images between two reference frames. A peak performance of 0.125 seconds for a 640x480 image can be achieved on a Nvidia Titan X (Pascal). In this paper, we will refer to this approach by "Inter$_{\text{dain}}$" for convenience.
- **Inter$_{\text{nearest}}$: Nearest Neighbor Interpolation** is the most basic interpolation technique. The intermediate image is constructed by copying the temporally closest reference frame. It is used as a baseline method to compare the other approaches against it. In this paper, we will refer to this approach by "Inter$_{\text{nearest}}$" for convenience.

The publicly available open source implementations of the aforementioned techniques have been modified to provide compatibility and a uniform user interface. The encapsulation into docker images assures consistent behavior across systems without complex dependencies. They are available on: `https://github.com/EricKinner/InterpScImgDB`

3 Method

Since the domain of scientific visualizations considerably differs from real world images, it has to be evaluated if image interpolation techniques work in this context. For that, the comparison of the aforementioned approaches is most naturally done by comparing the interpolated results to ground truth images.

The similarity of two images is measured through metrics. A multitude of metrics have been developed and each of them has its own characteristics, which highlight certain properties (e.g. sharpness) and hide others. To prevent a bias caused by one of these properties, multiple metrics are used instead. They ideally cancel each others shortcomings out and provide comparability to other studies.

- **SSIM** - Structural Similarity Index [35]. A perception-based metric widely used in image processing. (range [0-1], where higher values are good; one is perfect response)
- **MS-SSIM** - Multiscale Structural Similarity Index [33]. A development of SSIM with multiple stages of subsampling. (range [0-1], where higher values are good; one is perfect response)
- **VIFP** - Visual Information Fidelity in Pixel Domain [30]. Part of the commercial FVQA (Fusion-based Video Quality Assesment) [17] by Netflix for video quality monitoring. (range [0-1], where higher values are good; one is perfect response)

- **MSE** - Mean Squared Error. Basic absolute image error metric used in many applications. (range [0-$255^2 \times width \times height$], where lower values are good; zero is perfect response)
- **PSNR** - Peak Signal to Noise Ratio. Used by image compression algorithms and derived from MSE ($10 \times log_{10}(255^2 \div MSE)$) (range (0-inf], where higher values are good)
- **UQI** - Universal Quality Index [34]. Predecessor of SSIM. (range [0-1], where higher values are good; one is perfect response)

It is to be expected that a given interpolation approach achieves different quality for different types of visualizations. For that, a variety of datasets were chosen so that common classes of scientific visualizations are represented. The testing size is of course not exhaustive, but it is large enough to evaluate the feasibility in this domain and deduce tendencies.

- **Asteroid** - Volume rendering of asteroid impact with changing turbulences. Published by Patchett et al. [28].
- **Middlebury** - Standard dataset for optical flow estimation and image interpolation by Baker et al. [3].
- **MPAS** - Model for Prediction Across Scales [26]. Static mesh with maps of atmospheric and oceanic currents.
- **Nyx** - Cosmological simulation data of contracting mass into galaxy structures, provided by Lukic [22].
- **Viscous Fingers** - Isosurface rendered example dataset of ttk library [32] [21], originally generated for the 2016 IEEE visualization contest [9].

The comparison of image interpolation techniques and any other complex system involves a multitude of variables and variations. The analysis can be performed over several dimensions. The presented results are therefore only a summary of the measurements that were conducted. The full set of the results is available in the repository (`https://github.com/EricKinner/InterpScImgDB`). For the creation of new data, generalized docker images provide comfortable interfaces and allow the validation and continuation of this work.

Feasibility. The first step towards the deployment of image interpolation techniques on scientific visualizations is the examination of its feasibility. For this, it has to be compared against suitable alternatives. Scientific image databases are usually not resampled. Leaving the frame rate untouched is equivalent to the nearest neighbor interpolation ($\text{Inter}_{\text{nearest}}$). The normalization with this baseline approach gives insight on the amount of improvement and ensures the interpolation is not worse than actually doing nothing. Note that normalized results are only comparable in the amount of improvement over the baseline approach and not over their absolute error values.

Background Substitution. The background of scientific visualizations is usually a plain color. This is a distinct difference to common images, which might negatively influence the quality of the interpolated frames. The color of the background usually does not hold any information, that is of interest to researchers. Therefore, if more favorable configurations lead to better interpolation results, it could simply be exchanged. Background substitution can be a low-cost improvement for the adaptation of image interpolation techniques on this domain of visualization.

To explore this, a set of different background images were generated synthetically. The background is substituted by the color value, which leaves translucent pixels with the original background.

- **none** - Original image background.
- **green** - Color, which is unique in the image.
- **colorGrad** - Gradient between two unique image exclusive colors.
- **black** - Uniform black background.
- **white** - Uniform white background.
- **grad** - Gradient between black and white.
- **grid** - Original image background with rectangular white grid.
- **photo** - Photograph of real world scene with house and trees.

Fine-tuning. The majority of image interpolation techniques is based on convolutional neural networks as seen in the ranking of the middlebury dataset [3]. Fine-tuning is an established method to adapt neural networks to similar, but different domains of operation. Therefore, we examined how well fine-tuning can be used to adapt neural-network based image interpolation approaches to scientific visualizations. For this, the $\text{Inter}_{\text{dain}}$ method was trained on synthetic simulation-like data. Because quality and importantly, quantity is critical for successful machine learning, a collection of public and custom datasets were used. To extend the dataset with as many samples a possible, variations (e.g. volume, isosurface, etc.) of the underlying data were generated.

- **blender scene 1** - Custom blender scene of moving light sources and primitives. The intent with this dataset is to train the model on hard edges of solid meshes.
- **blender scene 2** - Custom blender scene of primitives, which emerge from each other and change in size over time. The intended effect of this dataset on the model is to introduce changes in mass and volume to the network. This is uncommon for real images, but a common occurrence in scientific simulations of e.g. isosurfaces or turbulences.
- **mrBrain** - Sliced scan of brain tissue, provided by [24]. Static frame with changing texture is meant to increase stability for semi-static scenes.
- **bonsai** - CT-Scan of a bonsai tree, provided by[11]. Rendered as volume with moving camera and changing isosurface.
- **lobster** - Resin embedded CT-Scan of a lobster, provided by [11]. Variations include moving volume rendering and changing isosurface.
- **csafe heptane** - Simulation step of combusting jet of heptane gas [11]. Training samples include volume renderings, sliced layers, and changing isosurfaces.

4 Results

Feasibility. The quality of a given image interpolation approach on scientific visualizations varies heavily. A positive example is displayed in Figure [1], which displays the $\text{Inter}_{\text{nearest}}$ normalized MSE error of $\text{Inter}_{\text{cylcic}}$ for all datasets. The scaling of the graph is below one. This indicates for the given normalized error metric, that the produced results are consistently better than the baseline approach. The least error is accumulated for the asteroid and nyx datasets. They even outperform the middlebury datset, which is the de-facto standard dataset for the comparison of image interpolation techniques. This confirms the general applicability of image interpolation on scientific visualizations. On some of the tested datasets an improvement of up to ten times could be achieved.

Yet for other approaches results like the ones displayed in Figure [2] can be observed. The normalized PSNR metric displays the lack of quality of $\text{Inter}_{\text{dain}}$ on all datasets. For this metric, values below one indicate worse quality in in comparison to the baseline ($\text{Inter}_{\text{nearest}}$). It can be seen that the interpolation of some datasets yield results worse than $\text{Inter}_{\text{nearest}}$. This technique is, therefore, not suited for the application on scientific visualizations.

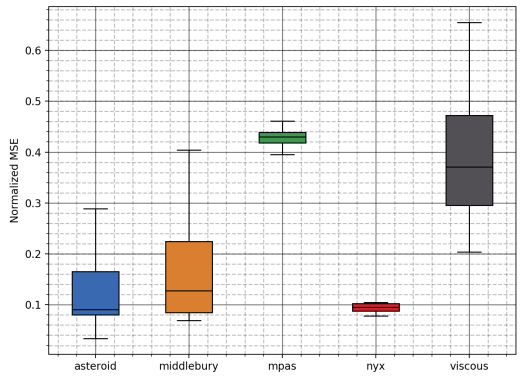


Figure 1 Evaluation of MSE error on single frame skipped interpolation by $\text{Inter}_{\text{cyclic}}$. Lower values are good (zero is perfect response).

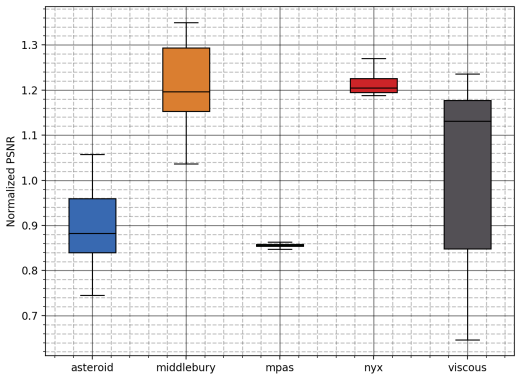


Figure 2 Evaluation of PSNR quality metric on single frame skipped interpolation by $\text{Inter}_{\text{dain}}$. Higher values are good.

In conclusion, it can be seen that the feasibility of general purpose image interpolation on scientific visualizations is heavily dependent on the dataset and method. Results range from much worse to up to ten times better than the baseline alternative ($\text{Inter}_{\text{nearest}}$). In the following we present suggestions for the selection of interpolation techniques. These aim to reduce the time to find an optimal configuration of data and approach.

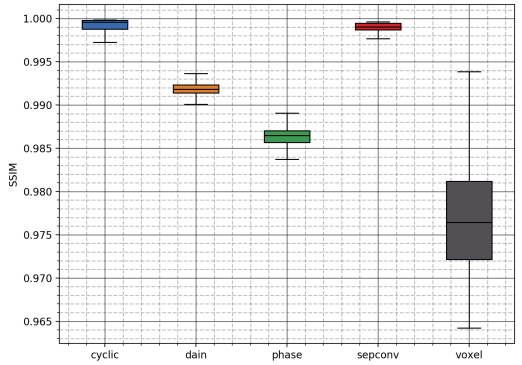


Figure 3 Evaluation of SSIM quality metric on the asteroid dataset with single frame skipped. Higher values are good. (one is perfect response).

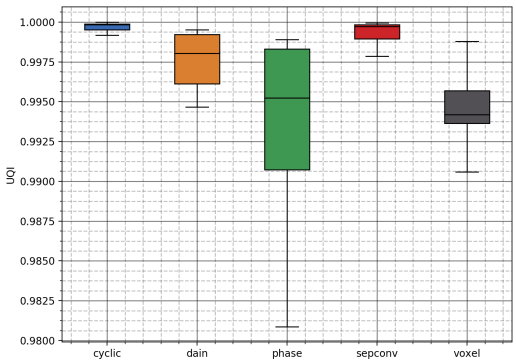


Figure 4 Evaluation of UQI quality metric on the asteroid dataset with single frame skipped. Higher values are good. (one is perfect response).

Limitations. The application of general purpose image interpolation techniques on scientific visualizations is in general feasible, but their results range a lot in quality. A closer look on their limitations and artifacts lead to a deeper understanding on favorable configurations.

For that, Figure [3] displays the quality of the interpolation approaches on the asteroid dataset in form of the SSIM metric. The asteroid dataset displays a volume rendered asteroid impact with subsequent atmospheric turbulences. A low quality and high variance of $\text{Inter}_{\text{voxel}}$ reveals its poor performance on volume-rendered images. Strict pixel- and voxel-flow based approaches enforce consistencies, which are not fulfilled in scientific simulations. There, colors are calculated through density functions of e.g. turbulences rather than being the product of previous motions.

The same configuration of dataset, approach, and frames skipped is displayed in Figure [4]. Here the the UQI quality metric is used. The $\text{Inter}_{\text{phase}}$ approach stands out with a high variance. A side-by-side comparison of the reconstructed frame with the original

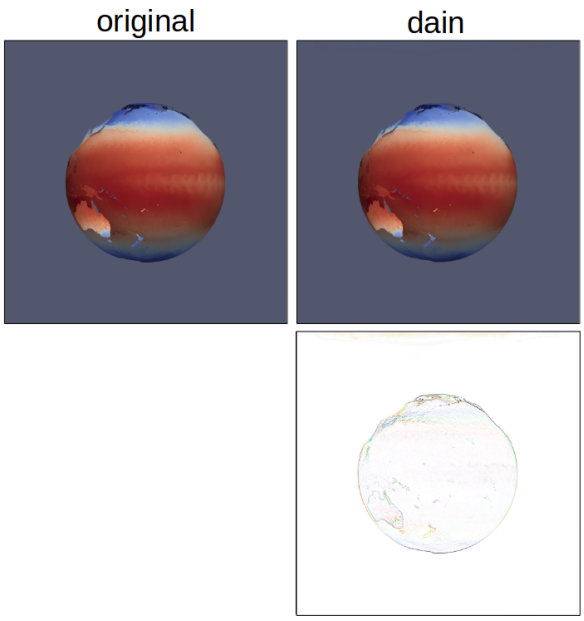


Figure 5 Difference from interpolated frame to original of the MPAS dataset with $\text{Inter}_{\text{dain}}$. Note: Enhanced contrasts for visibility.

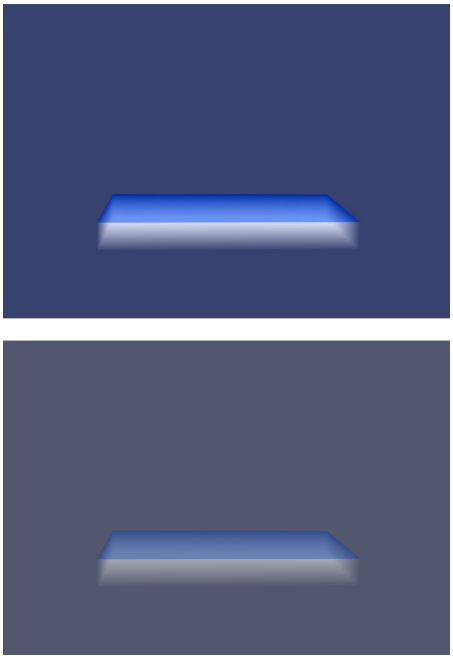


Figure 6 Interpolated frame (top) of the asteroid dataset with $\text{Inter}_{\text{phase}}$. Original frame (bottom) shows color difference.

one is shown in Figure [6]. It reveals that the colors of the interpolated frame are more saturated in comparison to the original. After the simulation progressed and a bigger portion of the screen is occupied by objects, the color difference becomes less distinct. Compression artifacts cannot be the reason for this anomaly, since lossless PNG compression [29] was used throughout the study. This suggests that the uniform colored background, which is most prominent in the first frames of the simulation, is the cause for this artifact.

A static mesh of the earth with changing textures of atmospheric and oceanic currents is displayed in the MPAS dataset. The $\text{Inter}_{\text{phase}}$ approach generates frames with slightly different color for this dataset as well. A different artifact is produced by $\text{Inter}_{\text{dain}}$. Because the general motion of the texture is directed towards a single direction, motion is falsely detected and corrected for the worse.. This causes the mesh to slightly change its position as a consequence. Figure [5] shows an example frame of the difference to ground truth. The fake motion causes a high error at the edges. Complex compound systems like $\text{Inter}_{\text{dain}}$ are capable of perceiving scenes very well. This generally increases the interpolation quality but may also lead to artifacts specific to scientific visualizations as the consistency of the scene is not necessarily preserved.

The Nyx dataset displays changing cosmological formations through sliced textures. The $\text{Inter}_{\text{voxel}}$ approach shows for this dataset a particular low quality. Figure [7] displays a interpolated frame of this dataset. It can be seen that the edge of the simulation region is fragmented and merged with the background. The temporal perception of the dataset is most probably related to this phenomenon. It displays the formation of cosmological structures. Through the accumulation of mass into stellar structures the texture seems to contract. The perceived contraction then causes $\text{Inter}_{\text{voxel}}$ to 'suck' pixels from neighboring regions (i.e. the background) into the simulation region. This did not occur for the motion of the MPAS

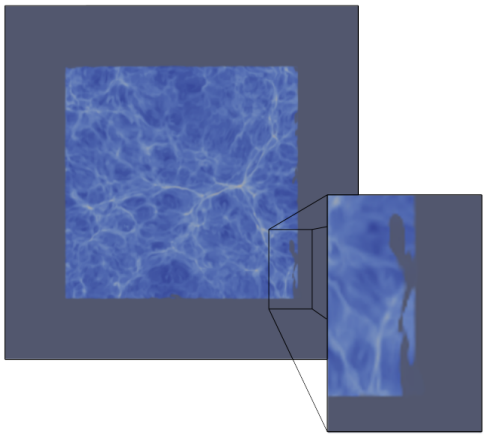


Figure 7 Interpolated frame of the nyx dataset, generated by $\text{Inter}_{\text{voxel}}$ approach. Areas on the edges are fragmented and merged with background.

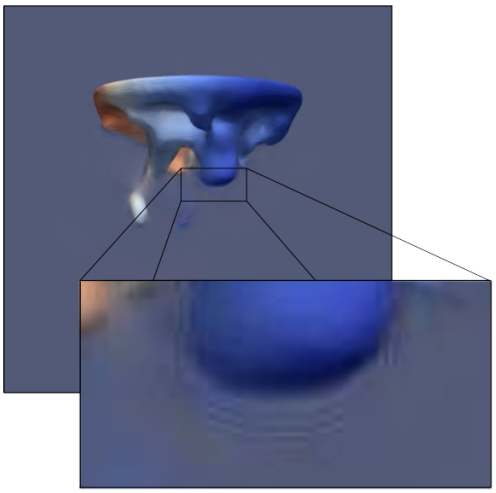


Figure 8 Interpolated frame of the viscous finger dataset, generated by $\text{Inter}_{\text{phase}}$ approach. Expanding fingers produce high frequency noise around the edges.

dataset. The reason might be the scale of the motion, which can only be perceived scene-wide and cannot be detected locally. This shows very clearly that strict flow-based approaches like $\text{Inter}_{\text{voxel}}$ are unable to deal with differences between optical flow and relative motion, which is not uncommon in scientific visualizations.

The most difficult dataset for all approaches is the viscous finger dataset. The isosurface rendering of mixing fluids (oil and water) shows expanding, contracting, and vanishing fingers, which appeared to be a challenge for most approaches. Each approach, however, reacted with different symptoms. The intermediate frames of $\text{Inter}_{\text{phase}}$ showed high frequency artifacts caused by the big ambiguity between consecutive frames like seen in Figure [8]. Other approaches, like $\text{Inter}_{\text{sepconv}}$ and $\text{Inter}_{\text{cyclic}}$, produced ghosting artifacts similar to linear blending. The $\text{Inter}_{\text{voxel}}$ interpolation shows the same artifacts as the ones created on the nyx dataset. The artifacts are clearly visible in Figure [9]. It shows well how the approach predicts paths from one image to another. It tries to fulfill the requirement of consistent voxel transitions. The fluctuations of mass is unlike any real world scenario though. This prevents matches between the frames, which are interpreted by most strict flow-based approaches as occlusion. And occluded areas are filled with a prediction of the background. The best reconstruction is from $\text{Inter}_{\text{dain}}$, which is capable of interpolating fingers with the least amount of blur. But it, too, is unstable on uniform backgrounds and produces fog-like artifacts in the beginning of the dataset as displayed in Figure [10]. Since this is the only dataset, where $\text{Inter}_{\text{dain}}$ showed these artifacts, it may be possible to adapt for it by fine-tuning, as seen later in the respective section.

Because every interpolation approach is different, it is not possible to make absolute statements about their applicability on scientific visualizations. But through domain knowledge and the conducted measurements, tendencies become apparent. With these recommendations simulation teams can exclude certain approaches and may save some time in the selection of suitable interpolation approaches.

Volume rendered images can be interpolated best by CNN based approaches, which are trained on optical flow, like $\text{Inter}_{\text{sepconv}}$ and $\text{Inter}_{\text{cyclic}}$. These methods are comparatively basic, but produce solid results for the nyx and mpas datasets as well. Algorithmic solutions,

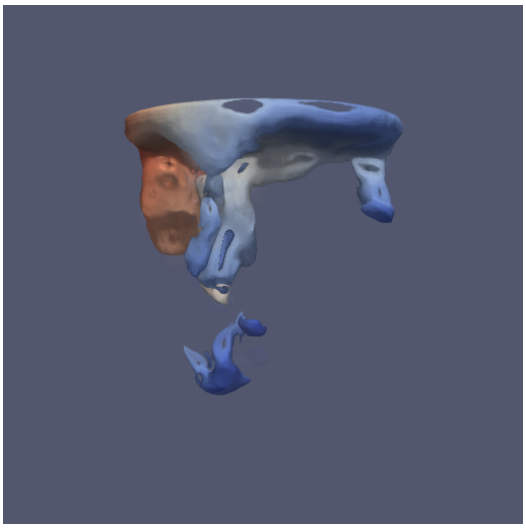


Figure 9 Frame interpolated by $\text{Inter}_{\text{voxel}}$ from viscous finger dataset.

Figure 10 Interpolated frame of the viscous finger dataset, generated by $\text{Inter}_{\text{dain}}$.

like $\text{Inter}_{\text{phase}}$, should be avoided, since they often rely on constraints and properties common in images but rare in scientific visualizations. Shown through multiple examples, it is also apparent that strict voxel or pixel flow based approaches produce heavy artifacts for changing masses and turbulences. Complex multi-component architectures can deal with these situations better since scene information is processed as well. But these systems are less predictable due to the high complexity and require careful fine-tuning to avoid a wrong perception of the scene.

The accuracy with respect to the desired features should be considered as well. If scientists want to derive other information from the interpolated images, stability with respect to e.g. classification, edge detection etc. has to be maintained. High-frequency noise like in Figure [8] or ghosting artifacts may invalidate the detection of edges, while it may be acceptable for a segmentation. This is also true for annotated data. Spatially coherent pixel-based information (like depth) can, in theory, also be interpolated and benefit from this approach. However, most available interpolation techniques are not designed for floating point images.

Iterative Application. The level of compression and maximum frame rate is directly linked to the number of intermediate frames. While some approaches allow for arbitrary interpolation between two reference frames, it is most common to interpolate a single frame in the middle of two reference frames. These approaches ($\text{Inter}_{\text{cyclic}}$, $\text{Inter}_{\text{voxel}}$, $\text{Inter}_{\text{sepconv}}$) have to be applied multiple times consecutively to achieve the same benefits. The number of intermediate frames through iterative application grows in $2^n - 1$. Approaches with variable number of intermediate frames ($\text{Inter}_{\text{dain}}$, $\text{Inter}_{\text{phase}}$) are more flexible in the selection of error thresholds for adaptive compression. In comparison, no definite qualitative difference between variable and iterative interpolation could be observed.

The maximum number of reconstructed frames is dependent on the initial sampling rate of the dataset. For very sparsely sampled data, image interpolation might not be a viable compression technique. But these datasets tend to benefit even more from the augmentation of their data and increased frame rates. The datasets, which were examined in this paper,

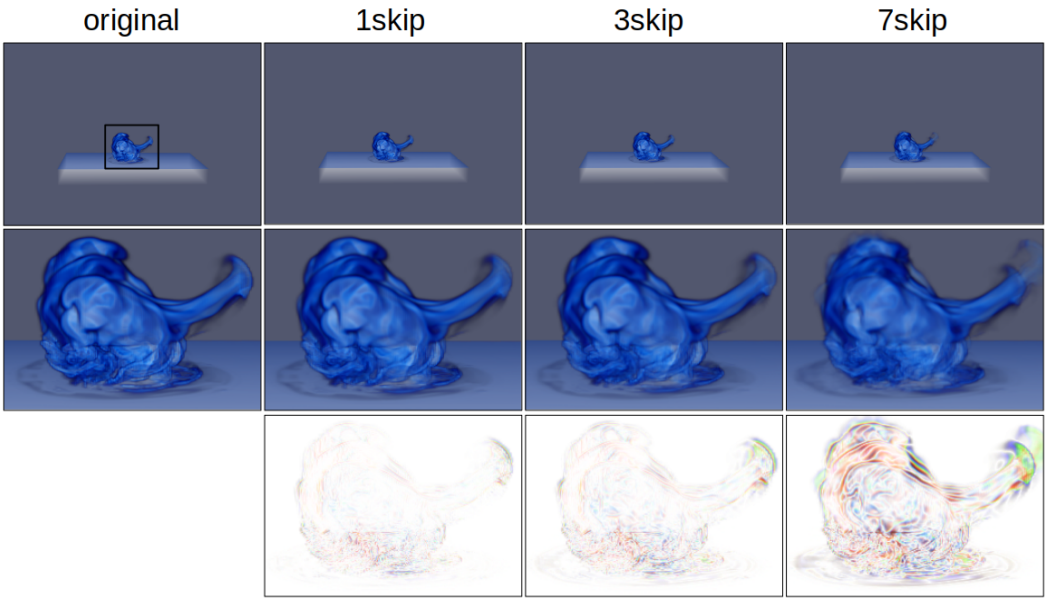


Figure 11 Comparison of example frame of asteroid dataset on $\text{Inter}_{\text{cyclic}}$ with iterative application and increasing interval of skipped frames.

had an average initial sampling rate. For them, we found empirically that 1-3 skipped frames result in acceptable quality. Only for densely sampled datasets 7 frames could be skipped (87% compression). Larger temporal steps between frames then tend to introduce aliasing artifacts. Figure [11] displays the results of iterative application of $\text{Inter}_{\text{cyclic}}$ on example frames of the asteroid dataset.

Table 1 Average performance of the interpolation approaches on a 512x512 frame inside the docker container. T_{init} is the time to load the model and all associated resources. T_{interp} is the time to interpolate a single frame. The measurements were conducted on Ubuntu 20.04.1 LTS, 16 GB RAM, i7-7700HQ @ 2.8GHz * 8, GeForce GTX 1050 Ti Mobile (4GB), Docker version 19.03.13.

approach	$Inter_{cylcic}$	$\text{Inter}_{\text{dain}}$	$\text{Inter}_{\text{phase}}$	$\text{Inter}_{\text{sepconv}}$	$\text{Inter}_{\text{voxel}}$
T_{init} [sec]	6.305	3.328	1.473	3.142	5.912
T_{interp} [sec]	2.248	23.841	5.352	0.220	3.520

Performance. The speed at which frames can be interpolated determines how the interpolation is used. Fast interpolations may be employed for online applications, while slower approaches would only be used for archiving data. Some of the approaches ($\text{Inter}_{\text{dain}}$, $\text{Inter}_{\text{cyclic}}$) use large models, which need to be loaded on initialization. Not only do they require sufficient GPU memory but also require significant time to load.

Table [1] displays the average performance on an 512x512 frame. T_{init} is the time needed for loading the model and other resources. T_{interp} is the time needed to interpolate a single frame. Note that the measurements were conducted inside a docker container. The virtualization overhead among other factors might influence the actual runtime. It is also important to note that the execution (T_{interp}) of $\text{Inter}_{\text{dain}}$ is most likely much lower. During the fine-tuning of $\text{Inter}_{\text{dain}}$ a processing speed of $\sim$1.22 seconds were observed. The training was conducted on an Nvidia GeForce GTX 2080 (8GB), which provided enough memory to

load all of the sub-models simultaneously.

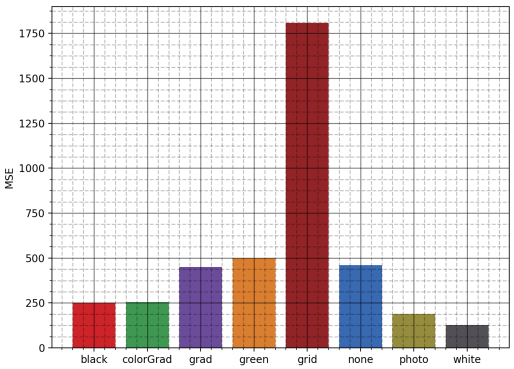


Figure 12 Evaluation of MSE error metric on a single frame of the asteroid dataset. The dataset was substituted with different backgrounds and interpolated by $\text{Inter}_{\text{phase}}$. Lower values are good. (zero is perfect response).

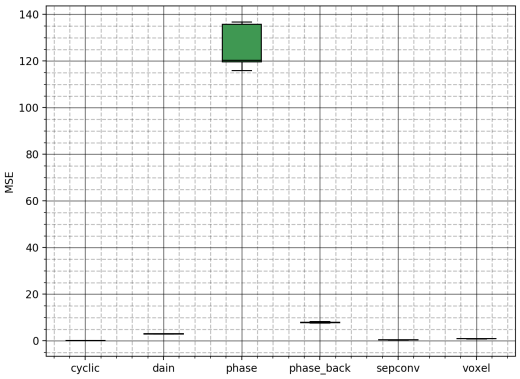


Figure 13 Evaluation of MSE error metric on the mpas dataset with a single frame skipped. Comparison of interpolation approaches including background substitution for $\text{Inter}_{\text{phase}}$. Lower values are good. (zero is perfect response).

Background Substitution. The background of scientific simulations can have a negative effect on the quality of the interpolation result. (see Figure [10] and [6]). The investigation and substitution can be of special benefit for algorithmic solutions ($\text{Inter}_{\text{phase}}$), since they generally lack other means of optimization. In the following, the effect of background substitution is examined on $\text{Inter}_{\text{phase}}$, which produced intermediate frames with slightly different color and contrast to the original dataset.

An example frame of the asteroid dataset is tested to select an optimal background for the $\text{Inter}_{\text{phase}}$ approach. Figure [12] displays the MSE error metric of the different substitutions. Contrary to expectations are the worst results produced by the grid background, which provides extra reference points. The uniform white background produced the best results for this approach. It also outperforms the substitution by the real world photo. This was not expected, since the methods were developed for real images. But this shows that well chosen synthetic images can actually be used to increase the quality of image processing methods. And since generally no extra information is encoded in background pixels, a substitution or initial selection of a favorable background can be a low-cost improvement for the later analysis framework.

The effect on the total quality of the reconstruction can be seen in Figure [13], which displays the results of all interpolation approaches aside the white substituted background. The former high error due to a change in colors is completely corrected and competitive quality can be achieved. A side-by-side comparison of frames from the MPAS dataset with the background substitution can be seen in Figure [17]. There, the improvement is not restricted to the correction of artifacts, but also reduces the error of the object. Background substitution can be applied to any approach regardless of their type of architecture. It reduces artifacts an improves the interpolation of the objects in the scene. It is a powerful tools for maximizing the quality of the reconstruction and is easily integrated into post hoc analysis frameworks.

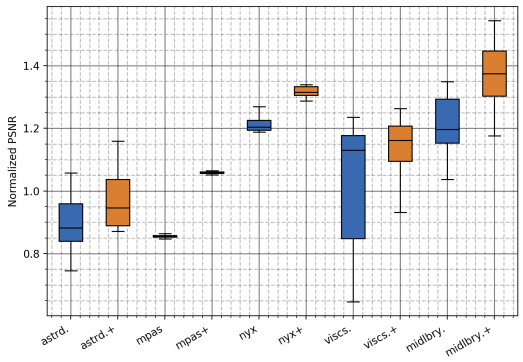


Figure 14 Comparison of fine-tuned $\text{Inter}_{\text{dain}}$ (indicated by "+")(orange) against the unmodified version (blue) on all datasets with PSNR quality metric. Higher values are good.

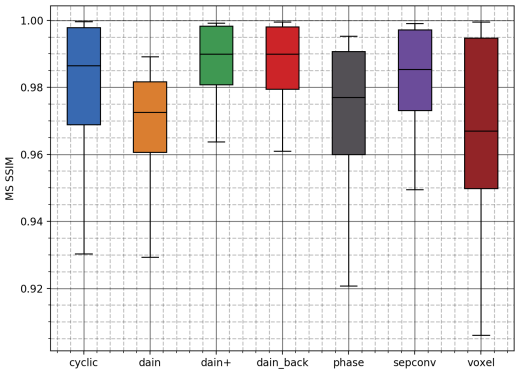


Figure 15 Evaluation of MS SSIM quality metric on the viscous dataset with 1 frame skipped and comparison to fine-tuned $\text{Inter}_{\text{dain}}$. Higher values are good. (one is perfect response).

Fine-tuning. The adaptation of machine learning approaches to new domains is commonly done via fine-tuning. In the following, $\text{Inter}_{\text{dain}}$ is fine-tuned. The new results give information about the magnitude of improvement, that can be expected from fine-tuning approaches or whether the architecture of the network is more important. The training was conducted on the provided model with default settings (apart from: num-epochs=32, lr=0.0005, batch-size=1) and on a Nvidia RTX 2080 with a total runtime of $\sim$ 6 hours.

The performance of the fine-tuned model (indicated by "+") in comparison to the unmodified approach is displayed in Figure [14] with the PSNR quality metric. The graph is normalized by $\text{Inter}_{\text{nearest}}$ to be able to display all datasets side by side. Higher values indicate better quality. The graphs shows that the quality increased for the fine-tuned version in comparison to the original model on all tested datasets. The extent of the improvement is also, as seen by the scale of the graph, of significant magnitude. The median increased, while variances decreased. Even the results for the middlebury dataset could be improved. Note that the metrics used in this evaluation are different from the metric used to determine the ranking of the middlebury dataset. But this shows that synthetic data can positively impact the training of conventional interpolation techniques. Artificially constructed edge cases like changing masses and deformations can deepen the understanding of the scene for the model and improve results. A frame-based side-by-side comparison of the fine-tuned version is shown in Figure [16].

The resulting performance of the fine-tuned model in comparison to all the other interpolation approaches, as well as a background substituted version, is shown in Figure [15]. The MS SSIM quality metric shows a significant increase in quality of the fine-tuned model. This changes its ranking among the other methods. $\text{Inter}_{\text{dain+}}$ is now the best choice. The rise in quality can mostly be attributed to the lack of artifacts, like seen in Figure [10], which is also achieved by suitable background substitution. Yet also edges and some fingers of the dataset are interpolated better, which could otherwise not been achieved. A side-by-side comparison is displayed in Figure [18], which illustrates the new robustness against moving textures, which previously caused errors like in Figure [5].

Although fine-tuning is able to overcome shortcomings on most datasets, it does not universally become the best option for every approach. The previous statements about the general performance of architectures on different datasets still hold. The correct choice of

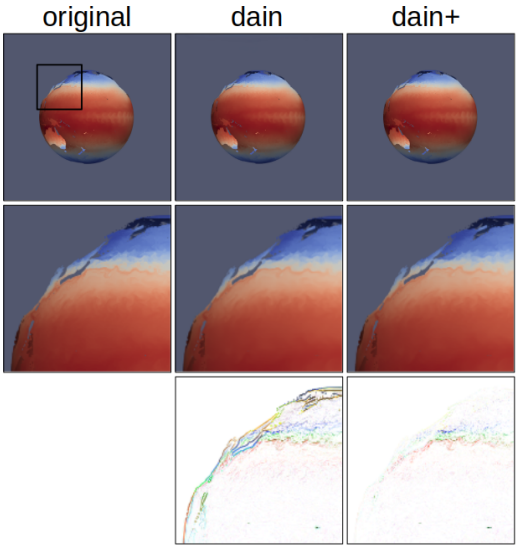


Figure 16 Comparison of example frame of asteroid dataset on $\text{Inter}_{\text{dain}}$ with fine-tuned version.

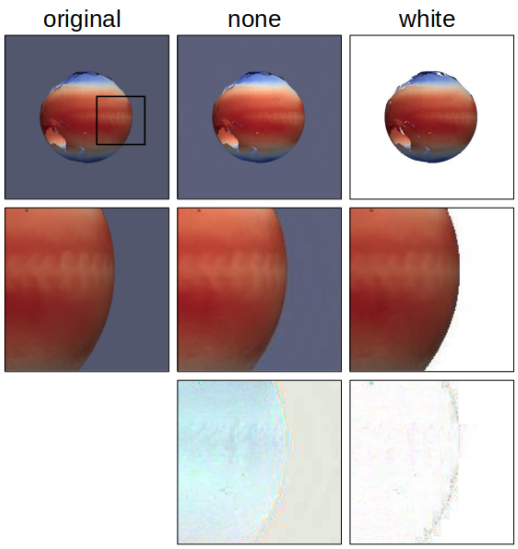


Figure 17 Comparison of example frame of mpas dataset on $\text{Inter}_{\text{phase}}$ with white background substitution.

architecture, visualization setting (i.e. background), and data type has a greater effect on the resulting quality. The application of image interpolation on scientific datasets without fine-tuning would a waste of potential. It is worthwhile and strongly recommended to adapt the interpolation approach in some way to the target dataset.

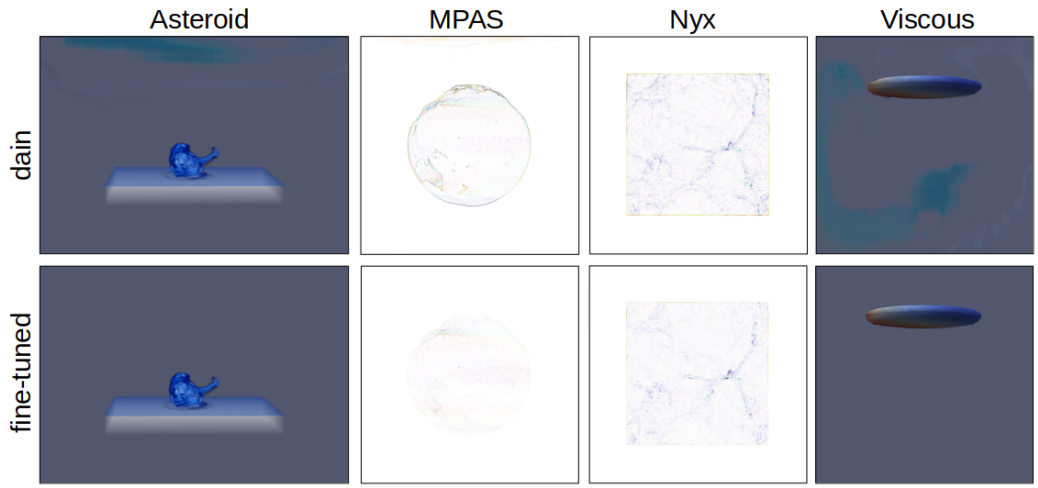


Figure 18 Example images of single frame skipped interpolation of all datasets between original and fine-tuned $\text{Inter}_{\text{dain}}$. MPAS and nyx dataset are displayed as enhanced difference images to increase the visibility of differences.

5 Conclusion

In this paper, we explored the feasibility of general purpose image interpolation techniques on scientific image databases. A survey over state-of-the-art methods with extensive evaluation over multiple metrics and datasets demonstrated the successful application on scientific visualizations. Tendencies toward favorable configurations could be observed, which give concrete recommendations for the selection of suitable interpolation approaches. A measurement of the runtime classifies the approaches for online application or post hoc data reduction. Invariably, all utilized approaches are publicly available through unified docker images. This enables this work to be verified or extended without setting up external dependencies. The limitations of e.g. iterative application have been examined and clarify how far and when the application of interpolation techniques is useful. Case studies for low-cost improvements, such as background substitution and fine-tuning, have shown to adjust systems for this domain of visualization.

In summary, image interpolation is a viable solution for improved user experience and storage compression of scientific image databases. Further research may include the development of expert systems for the interpolation of scientific interpolations as well as a composition of tools like view interpolation [20] and video compression [5] into a unified framework. The visualization of uncertainty for reconstructed frames would improve the acceptability and open its usage to more critical domains. Lastly, it may also be of interest to investigate the applicability of general purpose image interpolation onto volumetric datasets, which often are already sliced (e.g. MRT, CT) and could potentially be enhanced or compressed.

References

1 V. Adhinarayanan, W. c. Feng, D. Rogers, J. Ahrens, and S. Pakin. Characterizing and Modeling Power and Energy for Extreme-Scale In-Situ Visualization. In *2017 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 978–987, May 2017. doi:10.1109/IPDPS.2017.113.

2 James Ahrens, Sébastien Jourdain, Patrick O'Leary, John Patchett, David H. Rogers, and Mark Petersen. An Image-based Approach to Extreme Scale in Situ Visualization and Analysis. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '14, pages 424–434, Piscataway, NJ, USA, 2014. IEEE Press. doi:10.1109/SC.2014.40.

3 Simon Baker, Daniel Scharstein, J. P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, March 2011. https://vision.middlebury.edu/flow/eval/. doi:10.1007/s11263-010-0390-2.

4 Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-Aware Video Frame Interpolation. *CoRR*, abs/1904.00830, 2019. Implementation from https://github.com/baowenbo/DAIN. arXiv:1904.00830.

5 Anne S. Berres, Terece L. Turton, Mark Petersen, David H. Rogers, and James P. Ahrens. Video Compression for Ocean Simulation Image Databases. In Karsten Rink, Ariane Middel, Dirk Zeckzer, and Roxana Bujack, editors, *Workshop on Visualisation in Environmental Sciences (EnvirVis)*. The Eurographics Association, 2017. doi:10.2312/envirvis.20171104.

6 Martin Burtscher and Paruj Ratanaworabhan. FPC: A High-Speed Compressor for Double-Precision Floating-Point Data. *Computers, IEEE Transactions on*, 58:18–31, February 2009. doi:10.1109/TC.2008.131.

7 S. Di and F. Cappello. Fast Error-Bounded Lossy HPC Data Compression with SZ. In *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 730–739, 2016.

8 O. Fernandes, S. Frey, F. Sadlo, and T. Ertl. Space-time volumetric depth images for in-situ visualization. In *2014 IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV)*, pages 59–65, 2014.

9 IEEEVIS. Scientific Visualization Contest. http://www.uni-kl.de/sciviscontest/, 2016.

10 Timothy Jeruzalski, John Kanji, Alec Jacobson, and David I.W. Levin. Collision-Aware and Online Compression of Rigid Body Simulations via Integrated Error Minimization. *Computer Graphics Forum (Proc. SCA)*, 2018.

11 Pavol Klacansky. Open SciVis Datasets, December 2017. https://klacansky .com/open-scivis-datasets/. URL: `https://klacansky.com/open-scivis-datasets/`.

12 Anand Kumar, Xingquan Zhu, Yi-Cheng Tu, and Sagar Pandit. Compression in molecular simulation datasets. In Changyin Sun, Fang Fang, Zhi-Hua Zhou, Wankou Yang, and Zhi-Yong Liu, editors, *Intelligence Science and Big Data Engineering*, pages 22–29, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

13 Julian Kunkel, Michael Kuhn, and Thomas Ludwig. Exascale Storage Systems – An Analytical Study of Expenses. *Supercomputing Frontiers and Innovations*, 1(1), 2014. URL: `https://superfri.org/superfri/article/view/20`.

14 Sriram Lakshminarasimhan, Neil Shah, Stephane Ethier, Seung-Hoe Ku, C. S. Chang, Scott Klasky, Rob Latham, Rob Ross, and Nagiza F. Samatova. ISABELA for effective in situ compression of scientific data. *Concurrency and Computation: Practice and Experience*, 25(4):524–540, 2013. `doi:10.1002/cpe.2887`.

15 Juelin Leng, Guoliang Xu, and Yongjie Zhang. Medical image interpolation based on multi-resolution registration. *Computers + Mathematics with Applications*, 66(1):1–18, 2013. `doi:10.1016/j.camwa.2013.04.026`.

16 H. Li, Y. Yuan, and Q. Wang. Video Frame Interpolation Via Residue Refinement. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2613–2617, 2020.

17 J. Y. Lin, T. Liu, E. C. Wu, and C. . J. Kuo. A fusion-based video quality assessment (fvqa) index. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, pages 1–5, 2014.

18 Yu-Lun Liu, Yi-Tung Liao, Yen-Yu Lin, and Yung-Yu Chuang. Deep Video Frame Interpolation using Cyclic Frame Generation. In *Proceedings of the 33rd Conference on Artificial Intelligence (AAAI)*, 2019. Implementation from https://github.com/alex04072000/CyclicGen.

19 Ziwei Liu, Raymond Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video Frame Synthesis using Deep Voxel Flow. In *Proceedings of International Conference on Computer Vision (ICCV)*, October 2017. Implementation from https://github.com/liuziwei7/voxel-flow and https://github.com/lxx1991/pytorch-voxel-flow.

20 J. Lukasczyk, E. Kinner, J. Ahrens, H. Leitte, and C. Garth. VOIDGA: A View-Approximation Oriented Image Database Generation Approach. In *2018 IEEE 8th Symposium on Large Data Analysis and Visualization (LDAV)*, pages 12–22, 2018.

21 Jonas Lukasczyk, Gunther Weber, Ross Maciejewski, Christoph Garth, and Heike Leitte. Nested Tracking Graphs. *Comput. Graph. Forum*, 36(3):12–22, 2017. `doi:10.1111/cgf.13164`.

22 Zarija Lukic. Nyx Cosmological Simulation Data. http://dx.doi.org /10.21227/k8gb-vq78, 2019. `doi:10.21227/k8gb-vq78`.

23 Dhruv Mahajan, Fu-Chung Huang, Wojciech Matusik, Ravi Ramamoorthi, and Peter Belhumeur. Moving Gradients: A Path-Based Method for Plausible Image Interpolation. *ACM Trans. Graph.*, 28(3), July 2009. `doi:10.1145/1531326.1531348`.

24 Bill Lorensen Marc Levoy, Graham Gash. University of North Carolina Volume Rendering Test Data Set, 1987. URL: `https://graphics.stanford.edu/data/voldata/`.

25 S. Meyer, O. Wang, H. Zimmer, M. Grosse, and A. Sorkine-Hornung. Phase-based frame interpolation for video. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1410–1418, June 2015. Implementation from https://github.com/justanhduc/Phase-based-Frame-Interpolation. `doi:10.1109/CVPR.2015.7298747`.

26 MPAS. Model for Prediction Across Scales. https://mpas-dev.github.io/, 2019.

27 Simon Niklaus, Long Mai, and Feng Liu. Video Frame Interpolation via Adaptive Separable Convolution. *CoRR*, abs/1708.01692, 2017. Implementation from https://github.com/sniklaus/sepconv-slomo. `arXiv:1708.01692`.

28 John Patchett and Galen Gisler. Deep Water Impact Ensemble Data Set. Technical report, LANL, 2017. LA-UR-17-21595. URL: `https://datascience.dsscale.org/wp-content/uploads/2017/03/DeepWaterImpactEnsembleDataSet.pdf`.

29 PNG. Portable Network Graphics Specification. https://tools.ietf.org/html /rfc2083, March 1997.

30 H. R. Sheikh and A. C. Bovik. Image information and visual quality. *IEEE Transactions on Image Processing*, 15(2):430–444, 2006.

31 D. Tao, S. Di, Z. Chen, and F. Cappello. Significantly Improving Lossy Compression for Scientific Data Sets Based on Multidimensional Prediction and Error-Controlled Quantization. In *2017 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 1129–1139, 2017.

32 Julien Tierny, Guillaume Favelier, Joshua A. Levine, Charles Gueunet, and Michael Michaux. The Topology ToolKit. *CoRR*, abs/1805.09110, 2018. `arXiv:1805.09110`.

33 Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, volume 2, pages 1398–1402 Vol.2, 2003.

34 Zhou Wang and A. C. Bovik. A universal image quality index. *IEEE Signal Processing Letters*, 9(3):81–84, 2002.

35 Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.