DAGSTUHL
REPORTS

Volume 9, Issue 12, December 2019

ISSN 2192-5283

Published online and open access by
Schloss Dagstuhl – Leibniz-Zentrum für Informatik GmbH, Dagstuhl Publishing, Saarbrücken/Wadern, Germany. Online available at
http://www.dagstuhl.de/dagpub/2192-5283

Publication date
April, 2020

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 `http://dnb.d-nb.de`.

Digital Object Identifier: 10.4230/DagRep.9.12.i

Aims and Scope
The periodical *Dagstuhl Reports* documents the program and the results of Dagstuhl Seminars and Dagstuhl Perspectives Workshops.
In principal, for each Dagstuhl Seminar or Dagstuhl Perspectives Workshop a report is published that contains the following:
- an executive summary of the seminar program and the fundamental results,
- an overview of the talks given during the seminar (summarized as talk abstracts), and
- summaries from working groups (if applicable).

This basic framework can be extended by suitable contributions that are related to the program of the seminar, e. g. summaries from panel discussions or open problem sessions.

Contact
Schloss Dagstuhl – Leibniz-Zentrum für Informatik
Dagstuhl Reports, Editorial Office
Oktavie-Allee, 66687 Wadern, Germany
reports@dagstuhl.de
http://www.dagstuhl.de/dagrep

Report from Dagstuhl Seminar 19491

Big Graph Processing Systems

Edited by

Angela Bonifati[1], Alexandru Iosup[2], Sherif Sakr[3], and Hannes Voigt[4]

1 **University Claude Bernard – Lyon, FR,** angela.bonifati@univ-lyon1.fr
2 **VU University Amsterdam, NL,** alexandru.iosup@gmail.com
3 **University of Tartu, EE,** sherif.sakr@ut.ee
4 **Neo4j – Leipzig, DE,** hannes.voigt@neo4j.com

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 19491 "Big Graph Processing Systems". We are just beginning to understand the role graph processing could play in our society. Data is not just getting bigger, but, crucially, also more connected. Exploring, describing, predicting, and explaining real- and digital-world phenomena is increasingly relying on abstractions that can express interconnectedness. Graphs are such an abstraction. They can model naturally the complex relationships, interactions, and interdependencies between objects. However, after initial success, graph processing systems are struggling to cope with the new scale, diversity, and other real-world needs. The Dagstuhl Seminar 19491 aims to addresses the question: How could the next decade look like for graph processing systems?

To identify the opportunities and challenges of graph processing systems over the next decade, we met in December 2019 with circa 40 high-quality and diverse researchers for the Dagstuhl Seminar on Big Graph Processing Systems. A main strength of this seminar is the combination of the data management and large-scale systems communities. The seminar was successful, and addressed in particular topics around graph processing systems: ecosystems, abstractions and other fundamental theory, and performance.

Seminar December 1–6, 2019 – http://www.dagstuhl.de/19491

2012 ACM Subject Classification Computer systems organization → Architectures, Information systems → Data management systems, Software and its engineering → Distributed systems organizing principles

Keywords and phrases Abstractions, Big Data, Big Graph, data management, Ecosystems, graph processing, Performance, systems, Theory

Digital Object Identifier 10.4230/DagRep.9.12.1

1 Executive Summary

Alexandru Iosup (VU University Amsterdam, NL)
Angela Bonifati (University Claude Bernard – Lyon, FR)
Sherif Sakr (University of Tartu, EE)
Hannes Voigt (Neo4j – Leipzig, DE)

In memoriam: This seminar is dedicated to the memory of our co-organizer and friend Sherif Sakr (1979-2020), whose unexpected early departure happened a few months after the seminar. Sherif was a leading scientist in the field of Big Data Technologies. We are grateful to him for the time spent together and the joint work preceding and following the seminar. He will be deeply missed.

The world has become more interconnected than ever. Through an advancing wave of technologies and applications, our society is producing and consuming data at an unprecedented scale and complexity. To model the data, graphs offer a general model and mathematical abstraction, in the simplest form based on arbitrary objects (vertices) connected by relationships (edges), with possibly additional information (properties[1]). Graphs enable already a remarkable range of application domains[2], from industry to science, from society to governance, from education to gaming, but their true potential is just beginning to be unlocked. However, the tremendous increase in the size, complexity, and diversity of the graph-structured data and their applications, and the increasing community using graphs to understand and automate the world around us, raises new challenges for computer science. Under these new circumstances, the potential benefits of graph processing could be canceled by the difficulty to understand, create, develop, and automate graph processing for the masses. Focusing on the interplay between graph data, abstractions, systems, performance engineering, and software engineering, this seminar brings together researchers, developers, and practitioners actively working on this topic, to discuss *timely and relevant* open challenges with a main focus on the following topics: trade-off of design decisions of big graph processing systems, high-level graph programming abstractions and graph query languages, the specific requirements for different application domains for benchmarking and graph engineering purposes, systems and ecosystems for graph processing, the fundamental processes and methods leading to the science, design, and engineering of graph processing.

The seminar focused on the following key topics related to big graph processing systems:

Topic 1. Design Decisions of Big Graph Processing Ecosystems: In modern setups, graph-processing is not a self-sustained, independent activity, but rather part of a larger big-data processing ecosystem. Typical examples include the Giraph's deployment in the Facebook MapReduce ecosystem[3], Powergraph[4] in the GraphLab[5] machine learning and data-mining ecosystem, and GraphX[6] in the Apache Spark ecosystem. In general, more alternatives usually mean harder decisions for choice. In practice, with the wide spectrum of big graph processing systems, with different design decisions, that are currently available, it becomes very challenging to decide by intuition which system is the most adequate for a given application requirements, workload, or the underlying ecosystem. Making such decisions requires significant knowledge about the graph complexity, graph size, world requirements, and even the implementation details of the various available systems. Currently, we lack the fundamental models to understand and quantitatively analyze, estimate, and describe the complexity of big graph processing jobs. In addition, there is no understanding on the relationship between the graph complexity and the computational complexity of big graph processing jobs. Therefore, we need a clear understanding for the impact and the trade-offs of the various decisions (e.g., centralized vs distributed, partitioning strategy, programming model, graph representation model, memory storage vs disk storage) in order to effectively guide the developers of big graph processing applications.

[1] M. Junghanns et al., "Analyzing Extended Property Graphs with Apache Flink," NDA'16
[2] L. da Fontoura Costa et al. (2008) Analyzing and Modeling Real-World Phenomena with Complex Networks: A Survey of Applications, ArXiv Physics and Society, 2008. https://arxiv.org/abs/0711.3199v3 This study identifies tens of application domains for graph processing.
[3] Ching et al., One Trillion Edges: Graph Processing at Facebook-Scale, VLDB '15.
[4] Gonzalez et al., PowerGraph: Distributed Graph-Parallel Computation on Natural Graphs, OSDI '12.
[5] Low et al., Distributed GraphLab: a framework for machine learning and data mining in the cloud, VLDB '12.
[6] Gonzalez et al., GraphX: Graph Processing in a Distributed Dataflow Framework., OSDI '12.

Topic 2. High-Level Graph Processing Abstractions: While imperative programming models, such as *vertex-centric* or *edge-centric* programming models, are popular, they are lacking a high-level exposition to the end user. This way the end user is required more technical programming, which limits the end user productivity in building graph processing pipelines. In contrast, graph query languages build on more high-level, declarative constructs. Query language abstraction give more power to the less technical user and allow for extensive performance optimization by the underlying graph processing system. Current graph query languages, however, lack the power required in many graph analytics use cases. To increase the power of graph processing systems and foster the usage of graph analytics in applications, we need to design high-level graph processing abstractions. It is currently completely open how future declarative graph processing abstractions could look like, which the best level of abstraction is, how abstraction for analytics integrate with existing graph query languages, and we can evaluate new graph processing abstractions regarding utility, simplicity, expressiveness, and optimization potential.

Topic 3. Performance and Scalability Evaluation: Traditionally, performance and scalability are measures of efficiency, contrasting the ability of systems to utilize resources: FLOPS, throughput (e.g., EVPS), or speedup (i.e., compared to either a single-node, or a sequential implementation). Such metrics are difficult to apply for graph processing, especially since performance is non-trivially dependent on *platform*, *algorithm*, and *dataset* (i.e., the PAD triangle[7]). Therefore, many important questions arise: *how to compare the performance of graph-processing systems?, how to define scalability?, should one compare largely different systems, e.g., a distributed, heterogeneous system with a highly-tuned, hand-written sequential implementation?, how to design a framework for reproducible performance evaluation?*. Moreover, running graph-processing workloads in the cloud leverages additional challenges. First, we would like to understand whether the intrinsic cloud elasticity could be harnessed for graph processing. Second, clouds are known to be impacted by large degrees of performance variability due to colocation and virtualization overheads. Studying the impact of cloud performance variability onto graph-processing workloads is another topic of interest. Such performance-related issues are key to identify, design, and build upon widely recognized benchmarks for graph processing.

For each topic, the discussion also considered specific and general applications of graph processing, at various volume, velocity, and other dimensions.

The seminar brought together over 40 diverse and high quality researchers with core expertise from two generally distinct communities, data management and (large-scale) computer systems. The seminar was successful, and addressed in particular topics around graph processing systems: ecosystems, abstractions and other fundamental theory, and performance. To this end, we structured the seminar as follows:

1. Prior to the seminar, the co-organizers have contacted each participant, eliciting commitment for one or several topics, and ideas for key elements of the discussion.
2. During the first day of the seminar, the morning was dedicated to short presentations by each participant, and a long break-out session per topic. The former allowed the participants to better understand each other's core ideas and keywords, to identify synergies and to find experts for keywords not entirely familiar.
3. For the next two days, each morning challenged at least one half the participants with

[7] Guo et al., How well do graph-processing platforms perform? an empirical performance evaluation and analysis, IPDPS '14.

a tutorial given by a leading expert from the *other* community, then proceeded with break-out sessions organized per topic, and ended with a plenary session to share the main ideas. The tutorials were given by Tamer Özsu on “Graph Processing: A Panoramic View and Some Open Problems”, on behalf of the data management community, and by Antonino Tumeo on “Big Graph Processing: The System Perspective”, on behalf of the systems community. The main results of these two days of intense work were making terminology more uniform across the participants, and the core ideas about challenges (open problems), directions for long-term research, and identification of concrete short-term plans for continuation.

4. During the last day of the seminar, the participants finalized the immediate conclusions of the seminar (see Section “In Conclusion: Challenges and Future Directions for Big Graph Processing Systems”), and agreed on the plans for continuation.

2 Table of Contents

3 Overview of Talks

3.1 Graph Applications Focusing on Legal Cases at the Thomson Reuters Lab

Khaled Ammar (Thomson Reuters Labs, CA)

URL https://patents.google.com/patent/US20190347748A1/en

This talk presented work at Thomson Reuters Lab focusing on legal cases, around the question of "How to identify implied overruling for cited and citing legal cases?"

3.2 G-Core and Path Databases

Renzo Angles (University of Talca – Chile, CL)

Joint work of Renzo Angles, Marcelo Arenas, Pablo Barceló, Peter A. Boncz, George H. L. Fletcher, Claudio Gutierrez, Tobias Lindaaker, Marcus Paradies, Stefan Plantikow, Juan F. Sequeda, Oskar van Rest, Hannes Voigt
Main reference Renzo Angles, Marcelo Arenas, Pablo Barceló, Peter A. Boncz, George H. L. Fletcher, Claudio Gutierrez, Tobias Lindaaker, Marcus Paradies, Stefan Plantikow, Juan F. Sequeda, Oskar van Rest, Hannes Voigt: "G-CORE: A Core for Future Graph Query Languages", in Proc. of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018, pp. 1421–1432, ACM, 2018.
URL https://doi.org/10.1145/3183713.3190654

We present our current research around G-CORE, a proposal of query language for property graph databases. G-CORE was presented in SIGMOD'2018, and currently we are working in its implementation on top of Apache Spark. A novel feature of G-CORE is the use of paths as first class citizens. On this line, our current research concerns the notion of path database and the definition of operations for manipulating paths.

3.3 Native-Graph Native-Relational Query Support: The G+R (GRFusion) Approach

Walid Aref (Purdue University – West Lafayette, US)

Joint work of Mohamed S. Hassan, Tatiana Kuznetsova, Hyunchai Jeong, Walid G. Aref, Mohammad Sadoghi
Main reference Mohamed S. Hassan, Tatiana Kuznetsova, Hyun Chai Jeong, Walid G. Aref, Mohammad Sadoghi: "Extending In-Memory Relational Database Engines with Native Graph Support", in Proc. of the 21th International Conference on Extending Database Technology, EDBT 2018, Vienna, Austria, March 26-29, 2018, pp. 25–36, OpenProceedings.org, 2018.
URL https://doi.org/10.5441/002/edbt.2018.04

Graphs with relational node and edge schemas are popular. It is natural to use a relational system to support such graphs. However, this affects the performance of graph operations. In this talk, I will present the G+R Fusion Approach, where we can (1) seamlessly integrate graph and relational data, (2) natively process graph operations and natively process relational operations, (3) support declarative SQL-based graph and relational query formulation, (4) conceptually model a graph in SQL as a stream of vertexes, edges, paths, or subgraphs, (5) support complex graph operations via hints within SQL, (6) provide an API for programming complex graph operations from within SQL, (7) support graph as an in-memory index inside a relational engine with g2r and r2g pointers to facilitate the navigation between the relational and graph sides, (8) isolate graph topology from graph annotations and form graph index only on the topology, (9) introduce graph operations that co-occur in a relational query evaluation pipeline, e.g., vertexScan, edgeScan, and pathScan, (10) support complex queries that involve combinations of relational and graph operations both in native execution modes. The talk will report on in-progress work in support of Graph Views, SQL-based Graph-to-Graph transformations, and support for graphs with multiple node and edge types.

3.4 Understanding and Accelerating Graph Processing, Storage, and Analytics

Maciej Besta (ETH Zürich, CH)

Joint work of Maciej Besta, Torsten Hoefler, Marc Fischer, Tal Ben-Nun, Johannes de Fine Licht, Simon Weber, Lukas Gianinazzi, Robert Gerstenberger, Andrey Ivanov, Yishai Oltchik, Dimitri Stanojevic
Main reference Maciej Besta, Michal Podstawski, Linus Groner, Edgar Solomonik, Torsten Hoefler: "To Push or To Pull: On Reducing Communication and Synchronization in Graph Computations", in Proc. of the 26th International Symposium on High-Performance Parallel and Distributed Computing, HPDC 2017, Washington, DC, USA, June 26-30, 2017, pp. 93–104, ACM, 2017.
URL https://doi.org/10.1145/3078597.3078616

Combining theory and practice for understanding and accelerating graph processing, storage, and analytics, on all levels of the computing stack.

3.5 On the Linked Data Benchmark Council and Current Research

Peter A. Boncz (CWI – Amsterdam, NL)

Joint work of Peter A. Boncz, Renzo Angles, János Benjamin Antal, Alex Averbuch, Peter A. Boncz, Orri Erling, Andrey Gubichev, Vlad Haprian, Moritz Kaufmann, Josep-Lluís Larriba-Pey, Norbert Martínez-Bazan, József Marton, Marcus Paradies, Minh-Duc Pham, Arnau Prat-Pére
Main reference Renzo Angles, János Benjamin Antal, Alex Averbuch, Peter A. Boncz, Orri Erling, Andrey Gubichev, Vlad Haprian, Moritz Kaufmann, Josep-Lluís Larriba-Pey, Norbert Martínez-Bazan, József Marton, Marcus Paradies, Minh-Duc Pham, Arnau Prat-Pérez, Mirko Spasic, Benjamin A. Steer, Gábor Szárnyas, Jack Waudby: "The LDBC Social Network Benchmark", CoRR, Vol. abs/2001.02299, 2020.
URL https://arxiv.org/abs/2001.02299

This talk presents the past, present, and ongoing activities of the Linked Data Benchmark Council (LDBC), and state-of-the-art research in specializing Worst-Case Optimal Joins for graph processing, on updatable graph storage using Packed Memory Arrays, and the future of graph processing using the Spark framework.

3.6 Graph Metadata Management

Angela Bonifati (University Claude Bernard – Lyon, FR)

Joint work of Angela Bonifati, Peter Furniss, Alastair Green, Russ Harmer, Eugenia Oshurko, Hannes Voigt
Main reference Angela Bonifati, Peter Furniss, Alastair Green, Russ Harmer, Eugenia Oshurko, Hannes Voigt: "Schema Validation and Evolution for Graph Databases", in Proc. of the Conceptual Modeling – 38th International Conference, ER 2019, Salvador, Brazil, November 4-7, 2019, Proceedings, Lecture Notes in Computer Science, Vol. 11788, pp. 448–456, Springer, 2019.
URL https://doi.org/10.1007/978-3-030-33223-5_37

Despite the maturity of commercial graph databases, little consensus has been reached so far on the standardization of data definition languages (DDLs) for property graphs (PG). Discussion on the characteristics of PG schemas is ongoing in many standardization and community groups. Although some basic aspects of a schema are already present in most commercial graph databases, full support is missing allowing to constraint property graphs with more or less flexibility.

In this work, we show how schema validation can be enforced through homomorphisms between PG schemas and PG instances by leveraging a concise schema DDL inspired by Cypher syntax. We also briefly discuss PG schema evolution that relies on graph rewriting operations allowing to consider both prescriptive and descriptive schemas.

3.7 Some Thoughts on Graph Processing Systems

Khuzaima Daudjee (University of Waterloo, CA)

This talk addresses various design choices in graph processing systems, such as: which system issue or bottleneck to address, making data layouts graph-aware, and Scale-up vs. Scale-out. This discussion about graph processing leads to a reconsideration of classic distributed systems ideas.

3.8 Stream Reasoning: graph stream processing + AI

Emanuele Della Valle (Polytechnic University of Milan, IT)

Joint work of Emanuele Della Valle, Stefano Ceri, Frank van Harmelen, Dieter Fense, Daniele Dell'Aglio, Abraham Bernstein
Main reference Daniele Dell'Aglio, Emanuele Della Valle, Frank van Harmelen, Abraham Bernstein: "Stream reasoning: A survey and outlook", Data Sci., Vol. 1(1-2), pp. 59–83, 2017.
URL https://doi.org/10.3233/DS-170006

Stream and complex event processing studies the abstractions, the systems, and the applications to model, process, and manage data characterized by being highly dynamic and unbounded. This type of data is commonly named Data Stream or Event. Graph processing does the same for data characterized by a high variety and complexity. When I conceived Stream Reasoning [1], I placed it in the intersection of those two fields, to study the abstractions, the systems and the applications of inference techniques to streaming RDF data.

In more than 10 years of research [2], researchers active in Stream Reasoning proposed the streaming extensions of the Semantic Web stack as well as of other abstractions studied in Knowledge Representation, Artificial Intelligence, and Robotics. They also developed dedicated systems that find application in several settings, from Smart Cities to Industry 4.0, from the Internet of Things to Social Media analytics. I came to this Dagstuhl seminar on Big Graph Processing Systems to discuss the possibility to further study Stream Reasoning in the context of property graphs and learn from the works done in dynamic and streaming graph processing.

References

1 Emanuele Della Valle, Stefano Ceri, Frank van Harmelen, Dieter Fensel: It's a Streaming World! Reasoning upon Rapidly Changing Information. IEEE Intelligent Systems 24(6): 83-89 (2009) https://doi.org/10.1109/MIS.2009.125

2 Daniele Dell'Aglio, Emanuele Della Valle, Frank van Harmelen, Abraham Bernstein: Stream reasoning: A survey and outlook. Data Sci. 1(1-2): 59-83 (2017) https://doi.org/10.3233/DS-170006

3.9 Graph Query Processing Techniques

Stefania Dumbrava (ENSIIE – Evry, FR)

Joint work of Angela Bonifati, Stefania Dumbrava, Emilio Jesús Gallego Arias
Main reference Angela Bonifati, Stefania Dumbrava, Emilio Jesús Gallego Arias: "Certified Graph View Maintenance with Regular Datalog", Theory Pract. Log. Program., Vol. 18(3-4), pp. 372–389, 2018.
URL https://doi.org/10.1017/S1471068418000224

We have investigated two emerging aspects of graph query processing: 1) approximation over property graphs [SUM19] and 2) formal verification using the Coq proof assistant [TPLP18].

On the first line of research, we use an edge-centric approach to build query-driven quotient property graph summaries. In this context, we focus on the approximate evaluation of counting queries involving recursive paths (already known to be difficult to evaluate even over pure RDF graphs). We address this challenge with an in-database approach, whereby we perform both summarization, based on label-constraint reachability heuristics, as well as computation of relevant statistical information, stored as properties. We then translate queries from the original graph onto the summary. We experimentally assess the compactness of the obtained summaries and the accuracy of answering counting recursive queries on them.

On the second line of research, we used the Coq proof assistant to develop a mechanically-certified engine for incrementally evaluating and maintaining graph views. The language we use is Regular Datalog (RD), a notable fragment of non-recursive Datalog that can express complex navigational queries, with transitive closure as a native operator. To this end, we mechanize an RD-specific evaluation algorithm capable of fine-grained, incremental graph view computation, which we prove sound with respect to the declarative RD semantics. Using the Coq program extraction mechanism, we test an OCaml version of the verified engine on preliminary benchmarks, confirming the feasibility of obtaining a unified, machine-verified, formal framework for dynamic graph query languages and their evaluation engines.

3.10 Interoperability and Integration of Graph-Data Systems

Olaf Hartig (Linköping University, SE)

Being able to store, process, and analyze data about connections and relationships between things is important for scientists and businesses alike. While there exist a plethora of database management systems that specialize in such types of graph-based data, each of these systems is designed for a specific class of query and analysis features. Hence, for an organization to leverage its graph data for a broad range of use cases, it becomes advantageous to employ multiple such systems that complement each other. Unfortunately, most of the systems are incompatible in terms of how exactly they represent the graph data. As a consequence, these systems cannot easily be combined with one another.

The goal of my research is to establish the foundations of solutions that can be used to integrate graph data across across systems employed for maintaining and using such data. My approach to this end is to introduce formal abstractions based on which we can define and analyze concepts that are concerned with heterogeneous forms of graph data. Based on these abstractions, I am working on the following contributions: a) formal mappings between different models commonly used to represent graph data, b) results about fundamental properties of these mappings, c) formal tools to compare potential options for bridging different forms of graph data via the notion of virtual views, d) results shown based on this tools, and e) algorithms and techniques to employ these concepts in practice.

3.11 Big Graph Processing Challenges in Cryptoasset Analytics

Bernhard Haslhofer (AIT – Austrian Institute of Technology – Wien, AT)

Cryptocurrencies such as Bitcoin or Ethereum Token systems are well-known examples of cryptoassets. They build on blockchain technology and form virtual ecosystems in which different types of actors build and share economic relationships expressed in terms of transactions. Cryptoasset analytics aims at developing quantitative methods and tools that contribute to a better understanding of cryptoasset ecosystem. Such techniques are becoming increasingly relevant for a wide spectrum of use cases, ranging from scientific investigations, over compliance and regulation, to law enforcement. This talk will explain why big graphs processing is relevant for cryptoasset analytics, quickly outline the status quo, and point out technical challenges.

3.12 The Graphalytics Ecosystem

Tim Hegeman (VU University Amsterdam, NL)

Joint work of Tim Hegeman, Alexandru Iosup, Alexandru Uta, Wing Lung Ngai, Stijn Heldens, Arnau Prat-Pérez, Thomas Manhardt, Hassan Chafi, Mihai Capota, Narayanan Sundaram, Michael J. Anderson, Ilie Gabriel Tanase, Yinglong Xia, Lifeng Nai, Peter A. Boncz
Main reference Alexandru Iosup, Tim Hegeman, Wing Lung Ngai, Stijn Heldens, Arnau Prat-Pérez, Thomas Manhardt, Hassan Chafi, Mihai Capota, Narayanan Sundaram, Michael J. Anderson, Ilie Gabriel Tanase, Yinglong Xia, Lifeng Nai, Peter A. Boncz: "LDBC Graphalytics: A Benchmark for Large-Scale Graph Analysis on Parallel and Distributed Platforms", PVLDB, Vol. 9(13), pp. 1317–1328, 2016.
URL https://doi.org/10.14778/3007263.3007270

We present here work on the Graphalytics Ecosystem: From Competitions to Performance Analysis. Understanding how well do graph processing systems perform is important given the popularity of graph datasets, of graph processing techniques, and of graph processing systems aiming to process such datasets by combining such techniques. At the core of the Graphalytics Ecosystem is the LDBC Graphalytics benchmark, which in a nutshell is an advanced benchmarking harness, supporting many classes of graph-processing algorithms, with diverse real and synthetic datasets, a diverse set of experiments, and a renewal process to keep the workload relevant. LDBC Graphalytics enables comparison of many platforms, community-driven and industrial, under a set of well-defined metrics and with tools for validating results.

However suitable LDBC Graphalytics is for benchmarking and comparing graph-processing systems, we argue there is more to fully understand their performance. We define a design space of performance analysis tools, where depth of analysis (e.g., type and number of metrics) and breadth of experimentation (e.g., many and diverse experiments across many graph-processing systems) form the main dimensions. In this design space, the Graphalytics Ecosystem provides a *set* of complementary approaches for understanding graph processing performance, with the LDBC Graphalytics benchmark providing low depth and moderate breadth, the performance analysis instruments Grade10 and Granula providing much better depth but only for a narrow set of graph-processing systems, and the Graphalytics Global Competition provides the lowest depth but the broadest span of graph-processing systems.

3.13 Typing Graph Queries

Jan Hidders (Birkbeck, University of London, GB)

We study the problem of determining a query is well-typed given a certain graph schema. The graph schema is based on schemas as defined the Property Graph Schema Working Group. Moreover, we discuss typing regimes that define how to derive the type of a query if it is well-typed.

In addition we formulate an additional notion of schema based dependencies of the form type t1 is a subtype of type t2. We try to derive given a query, what is the weakest schema that guarantees that the query is well-typed.

3.14 Querying Knowledge Graphs on the Web

Katja Hose (Aalborg University, DK)

Numerous approaches and techniques for querying knowledge graphs have been proposed. They differ in their architectures (centralized, federated, cloud, etc.) and the problems they aim to solve (indexing, completeness, join processing, query optimization, provenance, semantic data warehousing, etc.). In particular with respect to knowledge graphs on the Web, there are many challenges that go beyond efficient query processing [1], e.g., encoding and querying metadata, overcoming heterogeneity, and deciding on the underlying paradigms of data storage (triple store vs. graph store) or query language (SPARQL vs. Cipher vs. GraphQL, etc.). But even the most advanced approach cannot deliver any results if the necessary data is (temporarily) not available. Hence, to solve this problem, we have recently developed an approach that builds on P2P systems to keep the data available [2] and that uses prefix-partitioned indexes [3] to efficiently identify relevant data.

References

1. G. Montoya, Hala Skaf-Molli, Katja Hose. *The Odyssey Approach for Optimizing Federated SPARQL Queries*. In Proceedings of the 16th International Semantic Web Conference (ISWC), 2017.
2. Christian Aebeloe, Gabriela Montoya, Katja Hose. *A Decentralized Architecture for Sharing and Querying Semantic Data*. In The Semantic Web – 16th International Conference (ESWC), 2019.
3. Christian Aebeloe, Gabriela Montoya, Katja Hose *Decentralized Indexing over a Network of RDF Peers*. In Proceedings of the 18th International Semantic Web Conference (ISWC), 2019.

3.15 Forecasting Information Diffusion in Online Environments: Lessons from DARPA's SocialSim Project

Adriana Iamnitchi (University of South Florida – Tampa, US)

This talk describes our graph processing experience while designing solutions for forecasting social media activities in various platforms, from GitHub software collaborations to cross-platform disinformation campaigns on Twitter and YouTube. Our experience is that while our raw datasets were huge, the graphs extracted in the end were relatively small, due to objective-specific data filtering (such as time windows of interests) and particularities of information diffusion phenomena (such as cascades of limited size). This observation highlights the need of connecting graph-processing components with (possibly existing, well established) general data processing components for facilitating graph extractions, data curation, etc.

3.16 The Distributed Systems Memex: Graph Processing Elements

Alexandru Iosup (VU University Amsterdam, NL)

Main reference Alexandru Iosup, Alexandru Uta, Laurens Versluis, Georgios Andreadis, Erwin Van Eyk, Tim Hegeman, Sacheendra Talluri, Vincent van Beek, Lucian Toader: "Massivizing Computer Systems: A Vision to Understand, Design, and Engineer Computer Ecosystems Through and Beyond Modern Distributed Systems", in Proc. of the 38th IEEE International Conference on Distributed Computing Systems, ICDCS 2018, Vienna, Austria, July 2-6, 2018, pp. 1224–1237, IEEE Computer Society, 2018.
URL https://doi.org/10.1109/ICDCS.2018.00122

In the 1940s, Vannevar Bush defined the concept of the personal *memex* as a person's device for storing and accessing all information and communication involving that person [1]. Bush identified many benefits for archiving large amounts of personal data into the memex, including learning about and eradicating diseases, enabling more creative and thought-related time by eliminating tasks that can be automated, etc. (Bush did not spend much time on the drawbacks, some of which are related to the appearance in the late-2010s of privacy-related regulations, e.g., GDPR.)

Similarly, we have posited in the early 2010s [2] that archiving large amounts of operational traces collected from the distributed systems that currently underpin our society into a Distributed Systems Memex can be beneficial for learning about and eradicating performance and related issues, for enabling more creative designs and extending automation, etc. We have added more recent concerns [3]: the Distributed Systems Memex could also serve as data-source to facilitate reproducible experiments, could preserve quantitative evidence for the future debates related to this class of systems, and could enable retrospective studies about the ethical operation of such systems. We also posit that a Distributed Systems Memex could help with the preservation of original designs and of their use, a valuable heritage.

Adding graph processing to the Distributed Systems Memex is non-trivial: What should the Distributed Systems Memex include? What data? Which types of distributed systems and ecosystems? How can such a Memex be designed, to support the concerns mentioned earlier? What instruments could it use and how could it be implemented overall?

The Distributed Systems Memex can only succeed as a community-driven effort.

References

1 Bush (1945) As we may think. The Atlantic, Jul 1945. [Online] Available: https://www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/
2 Iosup (2012) Towards logging and preserving the entire history of distributed systems. Is the Future of Preservation Cloudy? (Dagstuhl Seminar 12472), pp. 126–127. [Online] Available: https://doi.org/10.4230/DagRep.2.11.102
3 Iosup et al. (2019) The AtLarge Vision on the Design of Distributed Systems and Ecosystems. ICDCS 2019: 1765-1776. [Online] Available: https://arxiv.org/pdf/1902.05416.pdf

3.17 Representation learning of dynamic graphs

Vasiliki Kalavri (Boston University, US)

This talk introduces the problem of representation learning of dynamic graphs and discuss ideas towards avoiding retraining from scratch when changes occur. While inductive representation learning can generate predictions for unseen vertices, it cannot learn new knowledge from dynamic graphs. We describe a priority-based method for selecting rehearsed samples and show preliminary results on a classification problem.

3.18 Compiling graph algorithms to SQL

Hugo Kapp (Oracle Labs Switzerland – Zürich, CH)

In the past decades, relational databases have been filled up with huge amounts of data. Until now, SQL was seen as expressive enough to perform analytics on this data and answer specific questions. Nowadays, graphs are seen as a simple and intuitive way to express more complex analytics. To bridge the gap between these solutions, users have to export their data from their relational database(s), then import it into an external specialized graph processing engine.

To avoid moving data around, as well as to reduce the effort induced to reflect updates in the original relational data, we propose to perform some of these graph analytics directly on the relational data, inside the RDBMS. This requires the graph analytics workloads to be expressed in what RDBMSs can understand: SQL. The work presented here is focused on translating a procedural DSL for graph algorithms into SQL.

3.19 Analysis of Query Logs

Wim Martens (Universität Bayreuth, DE)

We present some of the results of a deep analysis of SPARQL query logs in collaboration with Angela Bonifati and Thomas Timm. We investigated about 500 million queries coming from Wikidata, DBpedia, geographic data, biological data, museum data, and analyzed a large range of their properties. In a first, shallow study we analyzed the queries size, keyword use, use of projection, etc. In a second, deeper study we analyzed the structure of queries, their (hyper)treewidth, and the structure of property paths (regular expressions) in the queries.

3.20 Towards Graph Processing Standards: The GQL Graph Query Language

Stefan Plantikow (Neo4j – Berlin, DE)

In this talk, we explore the language as the medium for graphs as the universal data model. We focus in particular on the "ISO/IEC WD 39075 – Database Language – GQL" standard, which aims to provide for property-graph databases a language specification that is next-generation, declarative, composable, compatible, modern, and intuitive.

3.21 Querying and Processing Big Property Graphs

Mohamed Ragab (University of Tartu, EE)

This talk introduces new ideas on querying and processing big property-graphs. We focus on the Morpheus framework as "Cypher on Top of Spark", which provides distributed Cypher queries and data integration for a variety of applications in data science. To this framework, we propose to add various graph-aware query optimizations techniques, and experiments leading to in-depth performance analysis.

3.22 BigGraph Projects at UBC

Matei R. Ripeanu (University of British Columbia – Vancouver, CA)

URL http://netsyslab.ece.ubc.ca/

We present here four Big Graph projects developed at UBC, in collaboration with many international partners. The projects are: Infrastructure for Graph Processing on Heterogeneous (CPU+GPUs) Platforms, Exact/ Approximate Pattern Matching for Massive Labelled Graphs, Infrastructure and Algorithms to Support Bi-Temporal Graph Exploration, and Infrastructure and Algorithms to Support On-Line Graph Analytics.

3.23 Graph Processing Systems Research Overview

Semih Salihoglu (University of Waterloo, CA)

We give a broad overview of the systems research graph processing that I work on. GraphflowDB is a graph database we are developing from scratch that rethinks many core components of a native graph database, including query processor, optimizer, physical design, indexes, and triggers. GraphSurge is a graph analytics system designed for applications that need to define multiple views of a base graph and run the same computation on each one. GraphWrangler is a software that is designed to extract graphs out of relational tables with two specific goals: (1) to obtain a graph view over relational tables with a few visual interactions, such as a drag and drops of rows and columns, and (2) to obtain a script that contains a mapping from relational tables to a property graph model, which can be used to perform ETL to move data from a relational system to a graph database.

3.24 Scalable Graph Algorithms and Partitioning

Christian Schulz (Universität Wien, AT)

Our research is mostly concerned with engineering scalable graph algorithms. This talk gives a broad overview and then focuses on one specific area: High Quality Graph Partitioning.

3.25 Querying Property Graphs

Petra Selmer (Neo4j – London, GB)

Property graphs have come of age in recent years. The querying of such graphs has naturally attracted a lot of interest and progressed considerably as a result.

One major development is the inception of GQL (Graph Query Language) as a new ISO/IEC project, with the aim of defining a comprehensive, expressive standardized declarative query language, which will, in the first instance, codify features provided by existing industrial property graph implementations. GQL will be substantially based on Cypher, which is implemented by Neo4j, RedisGraph, AnzoGraph, Memgraph, and SAP HANA Graph, as well as the Cypher for Apache Spark and Cypher for Gremlin projects. Although Cypher will form the substrate of GQL, there are other languages that will influence GQL, such as PGQL, G-CORE and the Property Graph Query Extensions currently under development in the latest version of SQL.

Collaborating with the University of Edinburgh, the formal denotational semantics of Cypher (both reading and updating semantics) have been defined. It is our aim to have in place a similar process for the specification of GQL, whereby formal semantics will inform and provide rigour in tandem with the production of the natural language specification that will be become the GQL Standard.

3.26 Designing and Building Enterprise Knowledge Graphs in Practice and My Research Interest

Juan F. Sequeda (data.world – Austin, US)

Joint work of Juan F. Sequeda, Willard J. Briggs, Daniel P. Miranker, Wayne P. Heideman
Main reference Juan F. Sequeda, Willard J. Briggs, Daniel P. Miranker, Wayne P. Heideman: "A Pay-as-you-go Methodology to Design and Build Enterprise Knowledge Graphs from Relational Databases", in Proc. of the The Semantic Web – ISWC 2019 – 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part II, Lecture Notes in Computer Science, Vol. 11779, pp. 526–545, Springer, 2019.
URL https://doi.org/10.1007/978-3-030-30796-7_32

Data Integration has been an active area of computer science research for over two decades. A modern manifestations is as Knowledge Graphs which integrates not just data but also knowledge at scale. Tasks such as Domain modeling and Schema/Ontology Matching are fundamental in the data integration process. Research focus has been on studying the data integration phenomena from a technical point of view (algorithms and systems) with the ultimate goal of automating this task.

In the process of applying scientific results to real world enterprise data integration scenarios to design and build Knowledge Graphs in practice, we have experienced numerous obstacles because the human and social factor is not taken into account. We argue that we need to think outside of a technical box and further study the phenomena of data integration with a human-centric lens: from a socio-technical point of view.

3.27 Building the Uber Graph

Joshua Shinavier (Uber Engineering – Palo Alto, US)

This short talk introduces Uber's knowledge graph and data standardization efforts, as well as the algebraic property graphs (APG) data model. APG is used for Uber's standardized schemas, as an intermediate language for transformations between schemas written in various other languages, and as a means of extending the graph processing idiom to diverse non-graph data sources. The data model has been formalized using category theory and with reference to the concept of algebraic databases, which is mentioned as a topic for additional future work.

3.28 Towards standardized benchmarks for graph processing

Gábor Szárnyas (Budapest Univ. of Technology & Economics, HU)

Defining benchmarks that capture certain types of workloads has been standard practice in database engineering. Such benchmarks offer a number of advantages: (1) they serve as case studies with openly available data sets and queries, (2) they capture a common understanding of what the systems-under-benchmark should be capable of, (3) they serve as a yardstick to compare the performance of industry tools, and (4) they free researchers from the burden of coming up with their own experiments, making their performance evaluations comparable to previous work.

Designing meaningful benchmarks is a major undertaking and necessitates a lengthy discussion to determine points of interest. It also requires creating a sizable software stack, consisting of a data generator, a driver that orchestrates benchmark executions, reference implementations, a reporting framework, etc. In the graph processing space, the Linked Data Benchmark Council (LDBC), established in 2012, has the goal of providing a standard benchmark suite that captures our current understanding of graph processing challenges. The current set of benchmarks consists of Graphalytics (focusing on graph analytics), the Semantic Publishing Benchmark (targeting RDF-based systems) and the Social Network Benchmark that focuses on graph queries (covering OLTP queries in the Interactive workload and OLAP queries in the Business Intelligence workload).

Our recent work was leading the Social Network Benchmark task force with the aim of extending the Business Intelligence workload by introducing delete operations and adding more queries that capture graph queries that are relevant for users but pose challenges for graph database systems.

3.29 Velocity on the Web

Riccardo Tommasini (Polytechnic University of Milan, IT)

Main reference Riccardo Tommasini: “Velocity on the Web – a PhD Symposium”, in Proc. of the Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, pp. 56–60, ACM, 2019.
URL https://doi.org/10.1145/3308560.3314192

The Web is a distributed environment populated by resources and agents that identify, represent, and interact with them. The decentralised nature of Web applications is one of the reasons for the popularity of the Web. Nevertheless, the Web results in an unbounded and noisy environment populated by heterogeneous resources. As part of the Web environment, applications must take resource heterogeneity into account. The Web of Data is the Web extension that addresses this challenge, known as Data Variety, using a stack of semantic technologies that include RDF, SPARQL, and OWL.

Recently, a new generation of Web applications is showing the need for taming Data Velocity, i.e., processing data as soon as they arrive and before it is too late. New protocols are emerging to improve the Web’s data infrastructure. Web Sockets and Server-Sent Events respectively enable continuous and reactive data access.

Data velocity is related to the whole data infrastructure, and new abstractions are required, i.e., streams and events that are the fundamental entities of the stream processing. Although seminal work on Stream Reasoning and RDF Stream Processing paved the road for addressing Velocity on the Web, the following research question remains unanswered: Can we identify, represent, and interact with streams and events on the Web?

3.30 HAGGLE and SO(DA)2: Two PNNL Graph System Projects

Antonino Tumeo (Pacific Northwest National Lab. – Richland, US)

This presentation briefly overviews HAGGLE: the Hybrid Attributed Generic Graph Library Environment, PNNL's project for the DARPA HIVE (Hierarchical Identify, Verify and Exploit) program, and SO(DA)^2: Software Defined Architectures for Data Analytics, one of the flagship project in PNNL's Data and Model Convergence Initiative. HAGGLE is designing stack to perform graph analytics on novel graph processors and exascale systems, implementing a Hybrid Data Model (graphs and tables). SO(DA)^2 is implementing a stack for Data Analytics targeting novel runtime reconfigurable architectures.

3.31 Serverless Graph Processing

Alexandru Uta (VU University Amsterdam, NL)

Joint work of Lucian Toader, Alexandru Uta, Ahmed Musaafir, Alexandru Iosup
Main reference Lucian Toader, Alexandru Uta, Ahmed Musaafir, Alexandru Iosup: "Graphless: Toward Serverless Graph Processing", in Proc. of the 18th International Symposium on Parallel and Distributed Computing, ISPDC 2019, Amsterdam, The Netherlands, June 3-7, 2019, pp. 66–73, IEEE, 2019.
URL https://doi.org/10.1109/ISPDC.2019.00012

Serverless computing promises ease-of-use, fine-grained scalability, elasticity and billing, and improved resource utilization via improved workload consolidation of many small functions instead of large monoliths. We investigate the benefits of serverless computing for graph processing workloads and implement the first serverless graph analytics prototype: Graphless. We show that Graphless is able to leverage the fine-grained elasticity that several analytics algorithms exhibit. However, to achieve high performance in a serverless environment, improved communication schemes and low-latency serverless storage.

3.32 Correlating graph properties with graph processing performance

Ana Lucia Varbanescu (University of Amsterdam, NL)

Joint work of Ahmed Musaafir, Alexandru Uta, Henk Dreuning, Ana-Lucia Varbanescu
Main reference Ahmed Musaafir, Alexandru Uta, Henk Dreuning, Ana-Lucia Varbanescu: "A Sampling-Based Tool for Scaling Graph Datasets". ACM/SPEC ICPE 2020 (accepted, in print)

It is well-known that, for many graph processing algorithms, performance is correlated with graph properties. However, there's little progress in analytically defining this correlations, even for well-studied algorithms like BFS or PageRank. In our work, we have attempted three different ways to better understand such correlation in the specific case of parallel graph processing algorithms. In this talk, we summarize our results in these three directions: (1) performance models based on hardware counters, (2) machine learning solutions for optimal algorithm selection, and (3) graph scaling to facilitate in-depth performance analysis. We illustrate the results we achieved in each direction with one relevant example, and list open questions that lie ahead.

3.33 High-Level Graph Processing Abstractions

Hannes Voigt (Neo4j – Leipzig, DE)

In this talk, we analyze the state-of-art in graph processing, and find that it is mainly driven bottom-up from hardware. We ask about better solutions: Are there even better abstractions? Where is the balance between expressivity and convenience (imperative vs. declarative)? How about the integration with query language: how big is the need, the desired degree, etc.? How could we systematically analyze commonality in graph programs? and, last but not least, What are optimization opportunities below an abstraction?

3.34 Graph! A Fundamental Structure for Maps

Hsiang-Yun Wu (TU Wien, AT)

URL http://yun-vis.net/

Visualization has been defined as a scientific field to take advantage of human vision and perception in order to amplify human cognition and gain insight in the complex data. Network visualization is one of the essential topics in the field to visually support the comprehensive understanding of the relationship in the underlying Big Data. More specifically, network layout, or namely graph drawing, has been considered as a key factor to understandability and memorability. In this talk, several network visualization techniques have been introduced to untangle nested and complex relationships using graphs, especially focusing on geospatial maps and biological pathway diagrams. Those maps are automatically computed through mimicking the design process of human illustrators. We extract meaningful aesthetic criteria and formulate them into machine computable mathematical equations. Simple examples, such as removing crossings through aligning two edges with minimum pairwise distances and uniformly distributing nodes over then entire map domain through centroid forces, are shown. Finally, the evaluation of usability is presented to qualitatively demonstrate the effectiveness of the proposed approaches.

3.35 AvantGraph: an open-source recursive analytics engine

Nikolay Yakovets (TU Eindhoven, NL)

We present AvantGraph, an upcoming, open-source graph processing system featuring main memory optimizations, recursive analytics, various vectorization and compilation advancements, and guarantees of worst-case optimality.

3.36 A Task-Centric Approach to Revolutionize Big Graph Systems Research

Da Yan (The University of Alabama – Birmingham, US)

URL https://cra.org/ccc/great-innovative-ideas/t-thinker-a-task-centric-framework-to-revolutionize-big-data-systems-research/

The short talk reviews existing works on Big Graph computational systems and Big Data computational systems in general, esp. those of the Hadoop Ecosystem, it indicates the weaknesses in existing research and envisions a new trend of compute-intensive system design that is able to fully utilize CPU cores in a computer cluster. A few Youtube videos on preliminary works are put online for those interested to check out at https://www.youtube.com/channel/UCQyivHtbwTIvwfO7ZWQVEow

3.37 Computer Systems Optimisation in Complex and Large Parameter Space

Eiko Yoneki (University of Cambridge, GB)

Managing efficient configurations is a central challenge for computer systems. I will introduce two recent projects: 1) Structured Bayesian Optimisation (SBO) to optimise systems in complex and high-dimensional parameter space, and 2) RLgraph, our framework for Reinforcement Learning (RL) to bring performance improvements to dynamically evolving tasks such as scheduling or resource management. These frameworks can be used tuning complex, high dimensional, and/or dynamic/combinatorial parameter space in computer systems. The applications that can take advantage of these optimisation frameworks are diverse ranging the hyper parameter tuning of neural networks, device placement of distributed execution engines (e.g. TensorFlow), transformation of computation graphs of Deep Neural Network, compiler optimisation, ASIC design, to database query optimisation.

4 Working groups

4.1 Working Group on Abstractions for Big Graph Processing

Renzo Angles (University of Talca – Chile, CL), Walid Aref (Purdue University – West Lafayette, US), Marcelo Arenas (PUC – Santiago de Chile, CL), Angela Bonifati (University Claude Bernard – Lyon, FR), Emanuele Della Valle (Polytechnic University of Milan, IT), Stefania Dumbrava (ENSIIE – Evry, FR), Olaf Hartig (Linköping University, SE), Jan Hidders (Birkbeck, University of London, GB), Hugo Kapp (Oracle Labs Switzerland – Zürich, CH), Wim Martens (Universität Bayreuth, DE), M. Tamer Özsu (University of Waterloo, CA), Stefan Plantikow (Neo4j – Berlin, DE), Sherif Sakr (University of Tartu, EE), Petra Selmer (Neo4j – London, GB), Juan F. Sequeda (data.world – Austin, US), Joshua Shinavier (Uber Engineering – Palo Alto, US), Hannes Voigt (Neo4j – Leipzig, DE), and Hsiang-Yun Wu (TU Wien, AT)

The purpose of this working group was to identify core requirements for graph processing abstractions. To this end, one has to consider the desiderata of practitioners, developers, and researchers alike and to balance implementational tractability with specification expressiveness. The main debate topics concerned defining a unifying foundation for graph models (the so-called "Dragon" model) and using this as the basis for achieving interoperability. First, we have singled out primitive graph operations that can serve as a comparison baseline and built a lattice of existing graph abstractions. Next, we have discussed constructive definitions of the top ("Dragon") lattice element, based on logics, algebra, type theory, and category theory. Finally, we have looked into centralized (star-shaped) and decentralized (peer-to-peer) approaches to navigating the lattice of abstractions and providing abstract mappings between this overarching specification and particular instances. We have also analysed the interoperability between the relational and graph paradigms, by looking at tipping point use cases.

Key elements of the discussion:
1. Human Data(base) Interaction,
2. Lattice of Abstractions,
3. Type algebras and Category Theory as a basis for a unifying foundation,
4. Logics as basis for a unifying foundation,
5. A graph/operational algebra,
6. Relational Meet Graphs.

4.2 Working Group on Performance of Big Graph Processing Systems

Alexandru Iosup (VU University Amsterdam, NL), Maciej Besta (ETH Zürich, CH), Peter A. Boncz (CWI – Amsterdam, NL), Khuzaima Daudjee (University of Waterloo, CA), Tim Hegeman (VU University Amsterdam, NL), Mohamed Ragab (University of Tartu, EE), Sherif Sakr (University of Tartu, EE), Semih Salihoglu (University of Waterloo, CA), Christian Schulz (Universität Wien, AT), Gábor Szárnyas (Budapest Univ. of Technology &

Economics, HU), Antonino Tumeo (Pacific Northwest National Lab. – Richland, US), Ana Lucia Varbanescu (University of Amsterdam, NL), Nikolay Yakovets (TU Eindhoven, NL), and Eiko Yoneki (University of Cambridge, GB)

The working group focused on the problem of assessing the performance of graph processing systems, i.e. how to design meaningful experiments and avoid reporting misleading results. *First*, we identified some key design decisions that have a significant impact on performance: Models (abstractions): data model, programming model, computational model, Architecture of systems, and Scalability considerations: load balancing, optimization, auto-tuning. These decisions are often intertwined, e.g. one can think of Pregel as computational model, as a system and as a reference architecture. *Second*, we identified some challenges for the systems and database communities w.r.t. the state-of-practice in assessing the performance of graph processing tools. The rest of the summary collects these.

Key elements of the discussion:

1. Methodological aspects of performance experiments,
2. Workloads,
3. Benchmarks, metrics and fine-grained instrumentation,
4. Reference architecture,
5. The Distributed Systems Memex,
6. Specialization vs. portability and interoperability,
7. Interplay between phenomena and complexity,
8. Design and performance space exploration,
9. Techniques to improve performance,
10. The gap between industry and academia/research,
11. Ambitious experiments on big graphs,
12. Vision for the next 5-10 years (+analysis of last 5-10 years).

4.3 Working Group on Ecosystems of Big Graph Processing Systems

Sherif Sakr (University of Tartu, EE), Khaled Ammar (Thomson Reuters Labs, CA), Angela Bonifati (University Claude Bernard – Lyon, FR), Khuzaima Daudjee (University of Waterloo, CA), Bernhard Haslhofer (AIT – Austrian Institute of Technology – Wien, AT), Katja Hose (Aalborg University, DK), Adriana Iamnitchi (University of South Florida – Tampa, US), Vasiliki Kalavri (Boston University, US), M. Tamer Özsu (University of Waterloo, CA), Eric Peukert (Universität Leipzig, DE), Matei R. Ripeanu (University of British Columbia – Vancouver, CA), Riccardo Tommasini (Polytechnic University of Milan, IT), Alexandru Uta (VU University Amsterdam, NL), Da Yan (The University of Alabama – Birmingham, US), and Eiko Yoneki (University of Cambridge, GB)

In modern setups, graph-processing is not a self-sustained, independent activity, but rather part of a larger big-data processing ecosystem with many system alternatives and possible design decisions. We need a clear understanding of the impact and the trade-offs of the various decisions in order to effectively guide the developers of big graph processing applications.

Key elements of the discussion:

1. Workloads of Graph Processing Ecosystems,
2. Overview/Vision/Standards/GQL,
3. Design Dimensions of Ecosystems,
4. (Populated) Reference Architecture,
5. Use Cases for Graph Processing Ecosystems,
6. System Requirements (Scale-up vs. scale-out),
7. Dynamic and Streaming Aspects,
8. Visualization Aspects (gaps between graph processing system and visualization, future technical challenge)

5 Panel discussions

5.1 Big Graph Processing: The System Perspective

Antonino Tumeo (Pacific Northwest National Lab. – Richland, US)

This tutorial introduces the terminology, core concepts, key solved and open challenges, and a roadmap for the future, taking a perspective from the (large-scale) computer systems community. We start by discussing how a big graph processing system is currently layered. We then go through abstractions, programming models, runtime, and custom architectures designs currently employed for Graph Processing. Finally, we discuss some of the ongoing research on Big Graph Processing System at PNNL, highlighting the research challenges, issues, and opportunities.

5.2 Graph Processing: A Panoramic View and Some Open Problems

M. Tamer Özsu (University of Waterloo, CA)

This tutorial introduces the terminology, core concepts, key solved and open challenges, and a roadmap for the future, taking a perspective from the data management community.

Participants

- Khaled Ammar
Thomson Reuters Labs, CA
- Renzo Angles
University of Talca – Chile, CL
- Walid Aref
Purdue University –
West Lafayette, US
- Marcelo Arenas
PUC – Santiago de Chile, CL
- Maciej Besta
ETH Zürich, CH
- Peter A. Boncz
CWI – Amsterdam, NL
- Angela Bonifati
University Claude Bernard –
Lyon, FR
- Khuzaima Daudjee
University of Waterloo, CA
- Emanuele Della Valle
Polytechnic University of
Milan, IT
- Stefania Dumbrava
ENSIIE – Evry, FR
- Olaf Hartig
Linköping University, SE
- Bernhard Haslhofer
AIT – Austrian Institute of
Technology – Wien, AT
- Tim Hegeman
VU University Amsterdam, NL
- Jan Hidders
Birkbeck, University of
London, GB
- Katja Hose
Aalborg University, DK
- Adriana Iamnitchi
University of South Florida –
Tampa, US
- Alexandru Iosup
VU University Amsterdam, NL
- Vasiliki Kalavri
Boston University, US
- Hugo Kapp
Oracle Labs Switzerland –
Zürich, CH
- Wim Martens
Universität Bayreuth, DE
- M. Tamer Özsu
University of Waterloo, CA
- Eric Peukert
Universität Leipzig, DE
- Stefan Plantikow
Neo4j – Berlin, DE
- Mohamed Ragab
University of Tartu, EE
- Matei R. Ripeanu
University of British Columbia –
Vancouver, CA
- Sherif Sakr
University of Tartu, EE
- Semih Salihoglu
University of Waterloo, CA
- Christian Schulz
Universität Wien, AT
- Petra Selmer
Neo4j – London, GB
- Juan F. Sequeda
data.world – Austin, US
- Joshua Shinavier
Uber Engineering –
Palo Alto, US
- Gábor Szárnyas
Budapest Univ. of Technology &
Economics, HU
- Riccardo Tommasini
Polytechnic University of
Milan, IT
- Antonino Tumeo
Pacific Northwest National Lab. –
Richland, US
- Alexandru Uta
VU University Amsterdam, NL
- Ana Lucia Varbanescu
University of Amsterdam, NL
- Hannes Voigt
Neo4j – Leipzig, DE
- Hsiang-Yun Wu
TU Wien, AT
- Nikolay Yakovets
TU Eindhoven, NL
- Da Yan
The University of Alabama –
Birmingham, US
- Eiko Yoneki
University of Cambridge, GB

Report from Dagstuhl Seminar 19502

Future Automotive HW/SW Platform Design

Edited by

Dirk Ziegenbein[1], Selma Saidi[2], Xiaobo Sharon Hu[3], and Sebastian Steinhorst[4]

1 **Robert Bosch GmbH – Stuttgart, DE,** dirk.ziegenbein@de.bosch.com
2 **TU Dortmund, DE,** selma.saidi@tu-dortmund.de
3 **University of Notre Dame, US,** shu@nd.edu
4 **TU München, DE,** sebastian.steinhorst@tum.de

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 19502 "Future Automotive HW/SW Platform Design". The goal of this seminar was to gather researchers and practitioners from academia and industry to discuss key industrial challenges, existing solutions and research directions in the design of future automotive HW/SW platforms, particularly focusing on predictability of systems regarding extra-functional properties, safe integration of hardware and software components and programmability and optimization of emerging heterogeneous platforms.

Seminar December 8–11, 2019 – http://www.dagstuhl.de/19502
2012 ACM Subject Classification Computer systems organization → Embedded software, Theory of computation → Models of computation, Software and its engineering → Real-time systems software
Keywords and phrases automotive, hw/sw platforms, real-time systems, systems design automation
Digital Object Identifier 10.4230/DagRep.9.12.28
Edited in cooperation with Lea Schönberger

1 Executive Summary

Dirk Ziegenbein (Robert Bosch GmbH – Stuttgart, DE)
Selma Saidi (TU Dortmund, DE)
Xiaobo Sharon Hu (University of Notre Dame, US)
Sebastian Steinhorst (TU München, DE)

Driven by new functionality and applications (such as automated driving and vehicle-to-X-connectivity) and fueled by the entry of new players from the IT industry, automotive systems are currently undergoing a radical shift in the way they are designed, implemented, and deployed. The trend towards automation and connectivity imposes an increased complexity and requires unprecedented computing resources, while, at the same time, the demanding requirements regarding cost-efficiency and dependability still need to be fulfilled. One of the most visible changes is the integration of formerly separated function domains onto centralized computing platforms. This leads to a heterogeneous mix of applications with different models of computation (e.g., control, stream processing, and cognition) on heterogeneous, specialized hardware platforms (comprising, e.g., application cores, safety cores, GPUs, deep learning accelerators) to accommodate advanced functionalities such as automated driving and on-line optimization of operating strategies for electrified powertrains.

The adoption of these novel heterogeneous platforms raises several challenges. In particular, many of their components stem from embedded consumer devices and have never been designed for application in safety-critical real-time systems. Therefore, while their computational capabilities are well understood, there is an increased need to comprehend these platforms from the perspective of extra-functional requirements such as predictability, determinism, and freedom-from-interference. This process deeply impacts the core design aspects of automotive E/E architectures and heavily challenges established methods and methodologies in HW/SW automotive design.

The goal of this Dagstuhl Seminar was to gather researchers and practitioners from academia and industry to discuss key industrial challenges, existing solutions and research directions in the HW/SW design of future automotive platforms. The seminar focussed, in particular, on

- predictability of systems regarding extra-functional properties,
- safe integration of hardware and software components and
- programmability and optimization of emerging heterogeneous platforms.

These inter-dependent challenges require the interaction between multiple disciplines, combining resource-constrained embedded, cyber-physical, and real-time aspects. Another important aspect of the seminar was to provide insight into novel automotive functionalities (such as automated driving, online optimization, or over-the-air-update) and their software architectures and requirements as well as into the HW/SW platforms they are executed on.

The seminar provided a unique opportunity for participants from the automotive industry to present their challenges and constraints and receive feedback and ideas from academia. At the same time, it allowed researchers to confront their own ideas and/or solutions with industrial reality and together identify new research directions in order to make an impact in the automotive industry.

Organization of the seminar

The seminar took place from 8th to 11th December 2019. The seminar started with an overview of current trends and challenges in the design of future automotive HW/SW platforms by the organizers. After that the agenda was structured along the previously mentioned challenges. Monday's talk sessions were focused on dependability and predictability of HW/SW systems. The sessions on Tuesday dealt with the safe integration of heterogeneous software applications covering aspects of software architectures, networks and cyber-physical systems in the automotive domain and touched societal issues as well. On Wednesday, the talks focused on the programmability and optimization of heterogeneous platforms. All talks were restricted to 15 minutes, leaving ample time for discussions as well as breakout sessions on the following topics:

- Modeling hardware and software dependencies
- Weakly hard real-time models
- Machine learning in cyber-physical systems
- HW/SW architecture exchange
- Benchmarking efforts for future HW/SW platforms
- Modularizing control systems
- Automotive software lifecycle
- Programming vs. execution models

More details on breakout sessions are available in a dedicated section of this document, after the overview of the talks given during the seminar.

Outcome

The seminar succeeded in bringing together participants from different communities who were engaged in very intensive, interdisciplinary group discussions. Not surprisingly, many participants stated that they were able to learn a lot from adjacent fields. As many of the industrial challenges at hand require interdisplinary approaches, the organizers consider this a significant success of the seminar. One example that became evident during the course of the seminar was that terms like execution model are quite differently used in e.g. the high performance computing domain and in the embedded systems community. A group formed in one of the breakout sessions intends to write a whitepaper on unifying terminology and formulating a common understanding of the different layers of models used in designing automotive HW/SW systems. A first follow-up meeting already took place in February 2020.

Several industrial presentations gave valuable insights in the industrial state-of-the-practice and outlined challenges for future research. A very good example for this was the breakout session "HW/SW Architecture Exchange" which discussed current architectural patterns and open challenges in the context of designing dependable systems and achieving deterministic behavior on heterogeneous high-performance HW platforms.

Another breakout session provided an overview of current automotive benchmarks and performance models that can be used as a basis for research activities. This session also raised the awareness that industry needs to be more active in providing relevant benchmarks in order to enable researchers to validate the industrial viability of their solutions.

Overall, the feedback of the participants showed that they made a lot of new contacts in academia and industry and a follow-up seminar in about two years was requested by many participants. The seminar inspired several new collaborations including contributions to the Autonomous Systems Design workshop at DATE 2020, ideas for special sessions at DAC 2020 and ESWEEK 2020 and also a student project on automotive HW/SW platform simulation between a students' project group and an industrial partner.

2 Table of Contents

Working groups

3 Overview of Talks

3.1 Towards a Contract Theory for Physical Systems

Bart Besselink (University of Groningen, NL)

Main reference Bart Besselink, Karl Henrik Johansson, Arjan van der Schaft: "Contracts as specifications for dynamical systems in driving variable form", in Proc. of the 18th European Control Conference, ECC 2019, Naples, Italy, June 25-28, 2019, pp. 263–268, IEEE, 2019.
URL http://dx.doi.org/10.23919/ECC.2019.8795736

Cyber-physical systems such as modern intelligent transportation systems generally comprise a large number of physical components connected through cyber elements for computation and communication. The large-scale, heterogeneous, and multi-disciplinary nature of these cyber-physical systems necessitates a theory for analysis, design, and control that is inherently modular, i.e., that allows for considering components independently. Whereas such theories exist for cyber elements in the form of contract theories, systematic approaches for physical components are lacking.

This talk will present some first results on a contract theory for physical systems by introducing assume/guarantee contracts for differential equation models of dynamical systems. The use of contracts as component specifications will enable modular design while providing guarantees on the behavior of the integrated system. The proposed framework is illustrated by application to an adaptive cruise control system and provides a first step towards a contract theory for cyber-physical systems.

3.2 Predictable Heterogeneous Computing for Next-generation Cyber-Physical Systems

Alessandro Biondi (Sant'Anna School of Advanced Studies – Pisa, IT)

Joint work of Alessandro Biondi, Marco Pagani, Francesco Restuccia, Giorgiomaria Cicero, Biruk Seyoum, Mauro Marinoni, Giorgio Buttazzo

Motivated by the increasing demand of high-performance computational capabilities to implement safety-critical functionality in next-generation automotive systems, this talk presents a series of research findings to enable predictability on heterogeneous computing platforms that integrate both asymmetric multiprocessors and hardware accelerators. The talk makes an opinionated point: FPGA-based system-on-chips are far more prone to predictability than other heterogeneous platforms. Such platforms have been found to particularly prone to enable time-predictable hardware acceleration (thanks to limited fluctuations of execution times and the possibility to control the bus/memory traffic with custom logic) and to design cost- and resource-efficient systems thanks to FPGA area virtualization. Such platforms are also sufficiently open to realize SW-based isolation mechanisms for SW tasks running on multiprocessors.

3.3 The Role of Programming Abstractions in Automotive Software

Jerónimo Castrillón-Mazo (TU Dresden, DE)

Digitalization and the new golden age for computer architecture pose great challenges for automotive software, as ever increasingly complex applications must safely execute on emerging computing systems. This talk argues for building strong semantics whenever possible in software platforms in order to keep this complexity under control. This is in contrast to more mainstream software approaches that, understandingly, focus on quickly providing functionality in a rather ad-hoc manner without a formal underlying execution model.

Formal models of computation represent a promising formalism to soundly frame automotive software design. Dataflow [1], for example, is a proven model that has enabled lots of methodologies for software and hardware synthesis [2]. Recently, methodologies have been extended to cope with the adaptivity required in embedded systems in general, and in automotive software in particular. By formalizing the structure of target parallel architectures and exploiting the semantics and the structure of dataflow programs [3], applications can be remapped dynamically at runtime while displaying superior time predictability [4]. The same formal properties of the model enable runtime decision-making to improve the energy efficiency of the overall system when multiple dataflow applications share resources [5]. All these methodologies and optimizations are possible only thanks to the properties of the underlying model of computation.

Adaptive extensions to dataflow methodologies are however not sufficient for automotive applications. Additional semantics are needed to *react* to external (sensory) inputs and to capture the inherent time dimension of cyber physical systems. Reactors [6] is a novel model that supports both reactive and timed behavior, enabling time determinism in a way similar to System-level LET [7]. We show how the reactor model can be enforced on top of the AUTOSAR Adaptive Platform via APIs, while still adhering to the standard. This way, we remove timing errors in the delivery and processing of frames in a video test application meant for automatic breaking [8]. In future work, we expect to be able to perform similar analysis and optimizations to these kinds of applications as done in the past for dataflow applications. Better language and platform-level support would also improve the programming experience.

Moving forward, extreme heterogeneity in the form of accelerators, novel post-CMOS fabrics and emerging memories will further complicate the process of mapping applications to hardware. Formal models and domain specific languages (DSLs) will become even more important to deal with such future systems, accompanied with software programming stacks that allow passing information up and down the stack [9, 10]. A tensor expression language [11], for example, could be used to pass information about required operations and a runtime resource negotiator could determine whether to provide access to acceleration in a tensor processing unit. Such tensor abstractions, made popular thanks to the machine learning boom, have already proven useful to optimize memory layout for emerging memory architectures in [12].

The golden age of computer architecture brings along a golden age for research on programming abstractions and software platforms. There is still a lot of work ahead, including efforts to refine the models to better match real automotive applications, to seamlessly integrate DSLs, and to improve information exchange within layers of the software platform.

References

1. J. B. Dennis. *First Version of a Data Flow Procedure Language.* In Programming Symposium, pages 362–376. Springer, 1974.
2. J. Castrillon and R. Leupers. *Programming Heterogeneous MPSoCs: Tool Flows to Close the Software Productivity Gap.* Springer, 2014.
3. A. Goens, S. Siccha, and J. Castrillon. *Symmetry in Software Synthesis.* ACM Transactions on Architecture and Code Optimization (TACO), 14(2):20:1–20:26, July 2017.
4. A. Goens, R. Khasanov, M. Hähnel, T. Smejkal, H. Härtig, and J. Castrillon. *Tetris: A Multi-application Run-time System for Predictable Execution of Static Mappings.* In Proceedings of the 20th International Workshop on Software and Compilers for Embedded Systems (SCOPES'17), SCOPES '17, pages 11–20, New York, NY, USA, June 2017. ACM.
5. R. Khasanov and J. Castrillon. *Energy-efficient Runtime Resource Management for Adaptable Multi-application Mapping.* In Proceedings of the 2020 Design, Automation and Test in Europe Conference (DATE), DATE '20. EDA Consortium, March 2020.
6. M. Lohstroh, Í. Romero, A. Goens, P. Derler, J. Castrillon, E. A. Lee, and A. Sangiovanni-Vincentelli. *Reactors: A Deterministic Model for Composable Reactive Systems.* In Proceedings of the 9th Workshop on Design, Modeling and Evaluation of Cyber Physical Systems (CyPhy 2019) and the Workshop on Embedded and Cyber-Physical Systems Education (WESE 2019), page 26pp, October 2019.
7. R. Ernst, L. Ahrendts, and K.-B. Gemlau. *System Level LET: Mastering Cause-effect Chains in Distributed Systems.* In IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society, pages 4084–4089. IEEE, 2018.
8. C. Menard, A. Goens, M. Lohstroh, and J. Castrillon. *Achieving Determinism in Adaptive AUTOSAR.* In Proceedings of the 2020 Design, Automation and Test in Europe Conference (DATE), DATE '20. EDA Consortium, March 2020.
9. J. Castrillon, M. Lieber, S. Klüppelholz, M. Völp, N. Asmussen, U. Assmann, F. Baader, C. Baier, G. Fettweis, J. Fröhlich, A. Goens, S. Haas, D. Habich, H. Härtig, M. Hasler, I. Huismann, T. Karnagel, S. Karol, A. Kumar, W. Lehner, L. Leuschner, S. Ling, S. Märcker, C. Menard, J. Mey, W. Nagel, B. Nöthen, R. Peñaloza, M. Raitza, J. Stiller, A. Ungethüm, A. Voigt, and S. Wunderlich. *A Hardware/Software Stack for Heterogeneous Systems.* IEEE Transactions on Multi-Scale Computing Systems, 4(3):243–259, July 2018.
10. G. Fettweis, M. Dörpinghaus, J. Castrillon, A. Kumar, C. Baier, K. Bock, F. Ellinger, A. Fery, F. H. P. Fitzek, H. Härtig, K. Jamshidi, T. Kissinger, W. Lehner, M. Mertig, W. E. Nagel, G. T. Nguyen, D. Plettemeier, M. Schröter, and T. Strufe. *Architecture and Advanced Electronics Pathways Towards Highly Adaptive Energy-efficient Computing.* Proceedings of the IEEE, 107(1):204–231, January 2019.
11. N. A. Rink, I. Huismann, A. Susungi, J. Castrillon, J. Stiller, J. Fröhlich, and C. Tadonki. *CFDlang: High-level Code Generation for High-order Methods in Fluid Dynamics.* In Proceedings of the 3rd International Workshop on Real World Domain Specific Languages (RWDSL 2018), RWDSL2018, pages 5:1–5:10, New York, NY, USA, February 2018. ACM.
12. A. A. Khan, N. A. Rink, F. Hameed, and J. Castrillon. *Optimizing Tensor Contractions for Embedded Devices with Racetrack Memory Scratch-pads.* In Proceedings of the 20th ACM SIGPLAN/SIGBED International Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES), LCTES 2019, pages 5–18, New York, NY, USA, June 2019. ACM.

3.4 Mixed Criticality Communication in Future In-Vehicle Architectures

Lulu Chan (NXP Semiconductors – Eindhoven, NL)

The autonomy and connectivity trends in the automotive industry lead to an increase in the number of functions and the amount of associated data in a vehicle. This talk presents an overview of the resulting challenges for automotive in-vehicle networks to help address these industry trends. This overview includes among others different architecture concepts, function mappings, and sharing of resources. Especially given the latter, there are challenges regarding how to ensure that the in-vehicle data communication is fast, prioritized, service oriented, safe and secure.

We review several solution approaches being discussed, such as recent TSN standards and the adoption of service oriented architectures.

The purpose of this talk is to trigger discussions on what the challenges and current solutions mean for future research directions.

3.5 Predictable and Reliable Automated Transportation Systems

Thidapat Chantem (Virginia Polytechnic Institute – Arlington, US)

Joint work of Thidapat Chantem, Ali Al-Hashimi, Pratham Oza, Thidapat Chantem, Ryan Gerdes
Main reference Ali Al-Hashimi, Pratham Oza, Ryan M. Gerdes, Thidapat Chantem: "The Disbanding Attack: Exploiting Human-in-the-Loop Control in Vehicular Platooning", in Proc. of the Security and Privacy in Communication Networks – 15th EAI International Conference, SecureComm 2019, Orlando, FL, USA, October 23-25, 2019, Proceedings, Part II, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Vol. 305, pp. 163–183, Springer, 2019.
URL http://dx.doi.org/10.1007/978-3-030-37231-6_9

As self-driving cars are becoming a reality, the dependability and reliability concerns for such systems lie at the forefront and must be addressed before a large-scale adoption can be expected. Up to now, we have relied on human operators to act as a "fail-safe" option, i.e., using human intervention to make critical decisions, when possible. However, systems have become so advanced and sophisticated that human intervention may in fact be detrimental. We will discuss such a case study in this talk, and to brainstorm ideas on design principles for the realization of safe systems of the future.

3.6 Specification-driven Design and Analysis for Perception, Decision-Making and Control in Autonomous Systems

Jyotirmoy Deshmukh (USC – Los Angeles, US)

Joint work of Adel Dokhanchi, Heni Ben Amor, Georgios Fainekos, Anand Balakrishnan, Aniruddh G. Puranic, Xin Qin

We argue that we can use partial specifications for components in self-driving cars to help reason about correctness and safety. We review the use of Signal Temporal Logic (STL) to express control-theoretic properties. We show how STL can be extended to a new logic called Timed Quality Temporal Logic (TQTL) that allows capturing sanity conditions on the results of perception. Such sanity conditions essentially encode geometric relations between objects detected in individual perception frames as well as physics-based laws governing how the representations of detected objects are temporally related across frames. We show [1, 2] how we can monitor such conditions on the outputs of state-of-the-art convolutional neural networks used in self-driving applications and discuss some results showing violations of such sanity conditions by such object detection CNNs.

References

1. A. Dokhanchi, H. B. Amor, J. V. Deshmukh, and G. Fainekos. *Evaluating perception systems for autonomous vehicles using quality temporal logic.* In International Conference on Runtime Verification, pp. 409-416. Springer, Cham, 2018.
2. A. Balakrishnan, A. G. Puranic, X. Qin, A. Dokhanchi, J. V. Deshmukh, G. Fainekos, *Specifying and Evaluating Quality Metrics for Vision-based Perception Systems.* In 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE) (pp. 1433-1438). IEEE.

3.7 Predictable Low-latency Data Services for Critical Applications – Challenges and Concepts

Rolf Ernst (TU Braunschweig, DE)

Joint work of Rolf Ernst, Leonie Köhler, Mischa Möstl, Kai Gemlau, Jonas Peeck
Main reference Rolf Ernst, Leonie Ahrendts, Kai Bjorn Gemlau: "System Level LET: Mastering Cause-Effect Chains in Distributed Systems", in Proc. of the IECON 2018 – 44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, October 21-23, 2018, pp. 4084–4089, IEEE, 2018.
URL http://dx.doi.org/10.1109/IECON.2018.8591550

Given the recent trend towards service oriented, data centric architectures at higher safety and real-time requirements, it is about time to re-evaluate the efficiency and appropriateness of the current in-vehicle communication. Using DDS as an example, the talk will highlight the upcoming challenges and discuss new concepts which combine all levels of a communication stack for superior efficiency. It will introduce a timed distributed programming paradigm that enables low-latency predictable IP communication stacks. The approach can be combined with application-level error handling to derive protected low latency object communication. The talk gave results and runtime measurements and outlined possible approaches to worst case end-to-end response time analysis.

3.8 Security and Correctness in the Face of Self-Adaptive Learning Automotive Systems

Sabine Glesner (TU Berlin, DE)

Automotive systems have changed in many ways during the last years. Not only their size has increased dramatically, also their interconnectedness has grown rapidly. Moreover, learning and adaptive components are finding their way into automotive systems. While it is well-known that safety is important in the design and implementation of automotive systems, nowadays also security has become a major issue. Since components are connected with each other and also with networks outside the car, intruders might find ways to corrupt the correct functioning of automotive systems. On top of this, learning and adaptive systems might change the functionality of the overall systems in unexpected ways. It is an open question what correctness means in the face of self-adaptive learning systems and how security can be ensured. In this talk, I will discuss these questions.

3.9 Aggregation and Integration of Next-generation Vehicle Computing and the OS Technologies

Masaki Gondo (eSOL – Tokyo, JP)

As the C.A.S.E, namely Connected, Autonomous, Sharing/Service, and Electric have become the driving force of the vehicle design, the E/E architecture design will also inevitably change. The design shift challenges the current highly distributed vehicle computing architecture, as the software needs to be aggregated and integrated to the level that is never imagined before. The software platform such as AUTOSAR or ROS with its underlying OS will be the key to realize a high-performance but energy-efficient, deterministic, cost-effective safe system. This talk will provide insights into the software challenges and the role OS plays to tackle the E/E architecture design shift.

3.10 Automotive Edge Computing Use-cases Inspired by Societal Problems

Baik Hoh (Toyota Motors North America – Mountain View, US) and Seyhan Uçar (Toyota Motors North America – Mountain View, US)

Joint work of Baik Hoh, Seyhan Uçar, K. Oguchi
Main reference Seyhan Uçar, Baik Hoh, K. Oguchi: "Management of Anomalous Driving Behavior". 2019 IEEE Vehicular Networking Conference, Los Angeles, USA, December 4-6, 2019.

Automotive HW/SW platform will soon undergo a radical shift to support future mobility trends, being connected, shared, electrified, and automated. Specifically, automakers have put an attention on connected cars for a sizable amount of time but surprisingly its definition still remains vague. Customers simply put CarPlay (or Android Auto) to a connected car while automotive industry engineers have struggled in realizing vehicles being connected to nearby vehicles, roadside unit, cloud, and nowadays edge. What makes it even worse is that every stakeholder perceives “edge” in a different way.

All these confusions must have been avoided if not a very product-oriented mindset (i.e., what technology can do) but a customer-centric approach (i.e., what future mobility and urban dwellers would need) called upon a connectivity and edge computing. Being motivated by key societal problems in urban mobility, Dr. Hoh and Dr. Uçar will share a few compelling use cases and insights in the first half. The second half will highlight the roles of emerging edge components to support the use cases. The industry inspired use case presentation would justify why vehicles should be connected to edge components. Also, the presented novel automotive functionalities would motivate an academic research community to include the emerging edge components in the equation of system verification and validation.

3.11 Paving the Way Towards Predictable Performance in Multi-heterogeneous SoC, Industrial Problems and Directions

Ignacio Sañudo (University of Modena, IT)

Joint work of Ignacio Sañudo, Marco Solieri, Micaela Verucchi, Marko Bertogna

Modern embedded platforms designed for automotive applications feature high-performance multi-core CPUs tightly integrated with compute accelerators. The spectrum of variety for such accelerators ranges from re-configurable devices and Graphic Processor Units for general purpose computing to Application Specific Integrated Circuits.

Many challenges have emerged with the integration of such technologies in the automotive domain. In this talk, novel solutions for improving the predictability of the future automotive hardware and software have been provided. For instance: (i) obscure scheduling mechanisms employed on GPUs impede deriving a proper timing analysis for applications. Understanding the scheduling decisions of modern GPUs is necessary to allow real-time engineers to safely model the performance of these accelerators – (paper under revision process) -. (ii) shared resources (such as caches and system RAM) allow latency depends on contention. Accurate memory-centric scheduling mechanisms for guaranteeing prioritized memory accesses to Real-Time components of the system might be a plausible solution to solve such problem [1]. (iii) engine heterogeneity brings another degree of complexity in the task partitioning problem. The Directed Acyclic Graph (DAG) model is appropriate to express the complexity and the parallelism of these tasks. Different aspects can be considered to convert a multi-rate DAG task-set with timing constraints into a single-rate DAG that optimizes schedulability, age and reaction latency, by inserting suitable synchronization constructs – (paper under revision process) –.

References

1 T. Kloda, M. Solieri, R. Mancuso, N. Capodieci, P. Valente, and M. Bertogna. *Deterministic Memory Hierarchy and Virtualization for Modern Multi-Core Embedded Systems.* In 2019 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS).

3.12 Possibilities Using FMI-based Co-simulation for the Validation of Cyber-Physical Systems

Peter Gorm Larsen (Aarhus University, DK)

Joint work of Peter Gorm Larsen, John Fitzgerald
Main reference Cláudio Gomes, Casper Thule, David Broman, Peter Gorm Larsen, Hans Vangheluwe: "Co-Simulation: A Survey", ACM Comput. Surv., Vol. 51(3), pp. 49:1–49:33, 2018.
URL http://dx.doi.org/10.1145/3179993
Main reference John S. Fitzgerald, Peter Gorm Larsen, Ken G. Pierce: "Multi-modelling and Co-simulation in the Engineering of Cyber-Physical Systems: Towards the Digital Twin", in Proc. of the From Software Engineering to Formal Methods and Tools, and Back – Essays Dedicated to Stefania Gnesi on the Occasion of Her 65th Birthday, Lecture Notes in Computer Science, Vol. 11865, pp. 40–55, Springer, 2019.
URL http://dx.doi.org/10.1007/978-3-030-30985-5_4
Main reference Casper Thule, Kenneth Lausdahl, Cláudio Gomes, Gerd Meisl, Peter Gorm Larsen: "Maestro: The INTO-CPS co-simulation framework", Simul. Model. Pract. Theory, Vol. 92, pp. 45–61, 2019.
URL http://dx.doi.org/10.1016/j.simpat.2018.12.005

The Functional Mockup Interface (FMI) has been invented in order to enable an open way to combine the simulation of different independent simulation units (FMUs) representing both cyber as well as physical parts of a Cyber-Physical System (CPS). The kay characteristics is that FMUs can be used in a black-box context such that suppliers are not revealing the IP captured in the model such FMUs are generated from. This presentation will demonstrate an open FMI-based tool chain called INTO-CPS that currently is supporting version 2.0 of the FMI standard, but where the team behind this also are heavily involved in the new version(s) of FMI attempting to ensure an even better support in the future.

3.13 Domain Controllers, Autonomous Driving and Functional Safety, Oh My!

Mark Lawford (McMaster University – Hamilton, CA)

Joint work of Victor Bandur, Vera Pantelic, Gehan Selim, Zinoviy Diskin, Nick Annable, Aalan Wassyng, Tom Maibaum, Asim Shah, and Spencer Deevy

With the advent of hybrid electric drive trains, advanced driver assistance (ADAS) and autonomous driving features being enabled by software, there was been a proliferation of Electronic Control Units and their software in modern vehicles. These new features together with existing automotive requirements for extensive product lines and regional requirements has resulted in an unsustainable growth of software development and safety costs [1]. We have seen in our research that OEMs are at the point where software development groups are required to produce a new release almost every workday to meet these needs.

How could this have happened? Let's consider a contrived example. In a hybrid electric vehicle you might currently have multiple Electronic Control Units (ECUs) connected via a Controller Area Network bus (CANbus) to implement the hybrid electric drive train functionality: the Engine Controller for the internal combustion engine, the Transmission Controller, possibly multiple (electric) motor controllers, a battery starter generator, a battery management unit, the body controller providing driver interfacing and a hybrid controller that orchestrates and optimizes the functioning of the power train. In the United States there is a regulatory requirement that the vehicle shall not roll away when keyed off and requiring the brake to be depressed before being able to move the vehicle out of park [2]. An OEM has a hybrid electric vehicle that meets this standard but now wants to sell the vehicle in South Korea where there a automated parking garages when unattended vehicles are moved around by automation so perhaps there is a market requirement that vehicle can be shut off and the keyed remove when in neutral so that the vehicle can be rolled for parking garages. Which ECUs require software change for our South Korean variant of the hybrid powertrain? Most likely the body control module that deals with the shifter and key interface and perhaps the transmission controller. We might also have some follow on functional safety change to the battery management unit to allow the car to be charged when the transmission is in neutral. There may also be some changes to software for the electric motor controls or battery starter generator. Thus a simple requirements change possibly requires 3 or more software releases (one for each affected controller).

One solution to this release proliferation problem is to go to a centralized architecture with new high speed networking technologies. In this case the hybrid controller implements the main power train functionality and the other systems effectively become smart actuators that respond to commands from the hybrid controller (eg. "up shift now" to the transmission controller, "provide this much torque" to electric motor controller. The hope is that when there is a requirements change only the hybrid controller as the power train domain controller would require a software update while the smart actuator ECUs would be left untouched.

Such a change in the electrical and electronic architecture of the vehicle raises some interesting questions:

- What are the functional safety implications of moving to a centralized architecture?
- How might the development and safety processes change to deal with them?
- What research problems must be solved to make these systems safe and cost effective, especially when the systems integrate machine learning components?
- How can virtualization be used to meet the requirements of running mixed criticallity software on a single powerful ECU?

In [3] we provide a survey of centralized automotive architecture research and development and discuss these research questions.

A large part of the cost of software intensive automotive system development is the effort required for functional safety and safety of the intended function. With so many vehicle variants across at typical OEM's product line it is impossible to create a separate complete safety case for each version of a vehicle. Improper reuse of safety related design artifacts can result in a vehicle recall so safety engineers currently are faced with the difficult and labour intensive task of determining what design artifacts need to be revised for a new vehicle design. Preliminary work on the use of model management techniques on explicit safety cases [4] has been proposed as a a way to help automate this process. The downfall of this technique is that assurance case formalisms such as Goal Structured Notation (GSN) lack sufficient detail of the dependencies of the evidence and how it is generated. In [5] and [6] we propose explicit modeling of the development and safety processes to provide fine grained traceability of safety artifacts that can allow model based incremental safety analysis that takes into account not only the safety artifacts but how they are produced. Some sort of rigorous, tool supported model based incremental safety analysis is going to be required for safe cost effective software development as systems become more complex and incorporate autonomous driving features. In addition standardized safety case templates for a given system is going to be required.

Autonomous vehicle architectures have typically implemented some form of the sense-plan-do framework with safety elements often integrated into the main architecture (e.g. [7]). While there have been works advocating for separation of safety and control [8], such works have not utilized well accepted safety patterns such as the 3-level safety of the eGas standard for throttle by wire [9]. We discuss our exploration of implementing this design pattern in autonomous vehicle safety architecture in [10], and in [11] provides a detailed exploration of safety architectures and associated assurance case templates for Level 5 autonomous vehicle.

References

1 R. Lancot. *Software is eating the auto industry.* Strategy CI Blog, Strategy Analytics, August 17, 2017. https://www.strategyanalytics.com/strategy-analytics/blogs/devices/strategic-ci/infotainment-telematics/2017/08/25/software-is-eating-the-auto-industry

2 NHTSA. Federal Motor Vehicle Safety Standard No. 114, Theft Protection and Rollaway Prevention, 74 FR 42837, National Highway Traffic Safety Administration, 2009.

3 V. Bandur, V. Pantelic, G. Selim, M. Lawford. *Towards Centralized E/E Vehicular Architectures.* Technical Report, McMaster Centre for Software Certification, (In preparation.).

4 S. Kokaly, R. Salay, M. Chechik, M. Lawford, and T. Maibaum. *Safety Case Impact Assessment in Automotive Software Systems: An Improved Model-based Approach.* In International Conference on Computer Safety, Reliability, and Security, pp. 69-85. Springer, Cham, 2017.

5 Z. Diskin, T. Maibaum, A. Wassyng, S. Wynn-Williams, and M. Lawford. *Assurance via Model Transformations and Their Hierarchical Refinement.* In Proceedings of the 21th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, pages 426-436, ACM, 2018.

6 Z. Diskin, N. Annable, A. Wassyng, and M. Lawford. *Assurance via Workflow+ Modelling and Conformance.* arXiv preprint arXiv:1912.09912, 2019.

7 K. P. Divakarla, A. Emadi, and S. Razavi. *A Cognitive Advanced Driver Assistance Systems Architecture for Autonomous-capable Electrified Vehicles.* In IEEE Transactions on Transportation Electrification **5**(1), pages 48–58, Mar. 2019.

8 C. B. Santos Tancredi Molina et al. *Assuring Fully Autonomous Vehicles Safety by Design: The Autonomous Vehicle Control (AVC) Module Strategy.* In 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W). Denver, CO, USA, pages 16-21, June 2017.

9 E-GAS Workgroup. Standardized E-Gas Monitoring Concept for Gasoline and Diesel Engine Control Units. Tech. rep. Version 6.0. July 2015.

10 S. A. Shah. *Safe-AV: A Fault Tolerant Safety Architecture for Autonomous Vehicles*, M.A.Sc. Thesis, Department of Computing and Software, McMaster University, Hamilton, ON, Canada, 2019.

11 S. Deevy. *Sentinel: Fault-Tolerant Software Architecture for Safe Artificial Intelligence in SAE J3016 Level 5 Autonomous Vehicles*, M.A.Sc. Thesis, Department of Computing and Software, McMaster, 2019.

3.14 Formal Verification on Finite-State Machines with Weakly-Hard Fault Models

Chung-Wei Lin (National Taiwan University – Taipei, TW)

Joint work of Chung-Wei Lin, Kai-Chieh Chang, Chao Huang, Shih-Lun Wu, and Qi Zhu

A weakly-hard fault model can be captured by an (m, k) constraint, where $0 \leq m \leq$ k, meaning that there are at most m bad events (faults) among any k consecutive events. We use a weakly-hard fault model to constrain the occurrences of faults. Given a constant K, we verify the system safety for all possible pairs of (m, k), where $k \leq K$, in an exact and efficient manner. By verifying all possible pairs of (m, k), we can analyze the optimal strategy to maximize resource saving (for system designers) or minimize attacking cost (for attackers). We believe that this is just an initial step, and many open problems will follow.

3.15 Sorry Software – Hardware Matters for Dependability

Albrecht Mayer (Infineon Technologies – München, DE)

Automated driving demands new levels of dependability (e.g. fail operational) and compute performance. Software complexity is traditionally addressed with approaches resulting in more memory and performance demands. But many software transparent hardware performance features contradict with predictability, dependability, security and hard real-time. High performance and high dependability at the same time can only be achieved with a carefully chosen system architecture, considering hardware failure rates, detection rates and recovery capabilities. The same is true for software which is usually even dominant in terms of failure rates. In this talk different compute components and system architectures will be assessed concerning their capabilities to deliver high dependable performance for hard real time systems.

3.16 Safe and Secure Software Platforms for Autonomous Driving

Philipp Mundhenk (Autonomous Intelligent Driving GmbH – München, DE)

URL http://www.aid-driving.eu

The vastly increasing amount of software in vehicles, its variability, as well as the computational requirements, especially for those built with autonomous driving in mind, require new approaches to the structure and integration of software. The traditional approaches of single-purpose embedded devices with integrated software are no longer a suitable choice. New architectures introduce general purpose compute devices, capable of high-performance computation, as well as high variability of software. Managing the increasing complexity, also at runtime, in a safe and secure manner, are open challenges. Solving these challenges is a high-complexity development and integration effort requiring design-time and runtime configuration, approaches to communication middleware, operating system configuration, such as task scheduling, monitoring, tight integration of security and safety, and, especially in the case of autonomous driving, concepts for dynamic adaption of the system to the situation, e.g., fail-operational concepts.

3.17 Design-For-Safety for Automotive IC Design: Challenges And Opportunities

Alessandra Nardi (Cadence – San Jose, US)

The autonomous driving revolution is introducing a whole new level of complexity in semiconductor design, not only in terms of computational/feature requirements but also because safety critical applications have very stringent demands on Functional Safety. However, the full automation supported in EDA (Electronic Design Automation) tools for traditional metrics has not yet reached maturity for the new safety metrics and is a green field of innovation. Additionally, the methodology and the criteria accepted for safety sign-off are still in evolution and in discussion in the industry. Even the language for capturing and exchanging the safety intent is yet to be fully defined. In this talk, we discuss some requirements to Design-For-Safety and some of the challenges and opportunities for flow automation that are presented to the semiconductor/systems community.

3.18 Dynamic Aspects of Centralized Automotive Software and System Architectures

Philipp Obergfell (BMW AG – München, DE)

Joint work of Philipp Obergfell, Jürgen Becker, Florian Ozwald, and Florian Sax
Main reference Simon Fürst, Markus Bechter: "AUTOSAR for Connected and Autonomous Vehicles: The AUTOSAR Adaptive Platform", in Proc. of the 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops, DSN Workshops 2016, Toulouse, France, June 28 – July 1, 2016, pp. 215–217, IEEE Computer Society, 2016.
URL http://dx.doi.org/10.1109/DSN-W.2016.24

Today's cars offer various control applications that may include the powertrain, the driving dynamics, or the characteristics of its body. Each of these control applications is realized by a series of electronic control units (ECUs) that interact on the basis of exchanged signals. Although these signals might be only relevant to the application layer of the ECU's software stack, also the low-level software needs to be defined statically whenever a new signal has to be either sent or received by an application.

From the perspective of OEMs, this development paradigm is no longer considered as reasonable. The main arguments in this respect are the increased set of signals (15.000 interactions among approximately 100 distributed ECUs) as well as the architecture's static definition which lacks in the support of light-weight updates.

One possible solution to overcome the mentioned disadvantages in current architectures are computing platforms which pursue three main goals: (1) The centralization of software applications from different existing ECUs, (2) the separation of applications and the low-level software by virtualization concepts, (3) and the interaction between applications on the basis of services.

As one precondition for their rollout, we consider two innovative modeling and model assessment methods. Both of them are related to the execution of software applications on component and system level and wrapped by the following questions:

1. How does the transition from statically defined software tasks in line with the AUTOSAR Classic Platform to POSIX processes and threads affect the design of computing platforms?
2. Which impact is illustrated by network paths between distributed software applications that are dynamically generated in line with the concept of runtime reconfiguration?

References

1 F. Oszwald, J. Becker, P. Obergfell, and M. Traub. *Dynamic Reconfiguration for Real-Time Automotive Embedded Systems in Fail-Operational Context.* IEEE International Parallel and Distributed Processing Symposium Workshops. 2018.

2 F. Oszwald, P. Obergfell, M. Traub, and J. Becker. *Using Simulation Techniques within the Design of a Reconfigurable Architecture for Fail-Operational Real-Time Automotive Embedded Systems.* IEEE International Symposium on Systems Engineering. 2018.

3 F. Oszwald,P. Obergfell, M. Traub, and J. Becker. *Reliable Fail-Operational Automotive E/E-Architectures by Dynamic Redundancy and Reconfiguration.* IEEE International System-on-Chip Conference. 2019.

4 P. Obergfell, S. Kugele, and E. Sax. *Model-Based Resource Analysis and Synthesis of Service-Oriented Automotive Software Architectures.* ACM/IEEE International Conference on Model Driven Engineering Languages and Systems. 2019.

5 P. Obergfell, F. Oszwald, M. Traub, and E. Sax. *Viewpoint-Based Methodology for Adaption of Automotive E/E-Architectures.* IEEE International Conference on Software Architecture Companion, ICSA Companion. 2018.

3.19 Metric-driven, System-level Testing: Release Autonomous Systems with Confidence

Maximilian Odendahl (Silexica – Köln, DE)

At Silexica, we're passionate about complex software solutions and solving the challenges associated with developing such systems. Today, we see that it is more and more difficult to keep the quality of the software stacks running. In the future, software complexity will continue to increase. Especially if you look at mobile systems like automated driving or robotics which have a high likelihood of doing harm, it is paramount that the software stacks within such systems are developed with rigor and thoroughness. Nobody wants to ship faulty products, right?

After investigating how modern software systems are developed and tested it became clear to us that there's a gap that needs filling. With the inherent complexity of today's software systems (and future systems even more so) traditional approaches of debugging and testing are not adequate to capture all defects. There are so many software components involved that when an error occurs, developers often spend several days or even weeks to figure out the root cause. Further, there is often no overview available which shows the complete system state including relevant metrics so that you can draw conclusions about where the root-cause might lie or determine if there's a new defect forming in the system.

With SLX Analytics, Silexica has created a new disruptive testing and integration platform that helps solving those challenges by providing automated testing of non-functional system-level metrics. It offers mult-level insights into all relevant Software Layers at once (from application, through middleware down to OS layer), allowing you to drill-down into the system state at the failing point in time. This helps to minimize the time spent on root-cause analysis and to detect (hidden) system defects in your products.

3.20 Parallel Programming Models for Critical Real-time Embedded Systems

Eduardo Quinones (Barcelona Supercomputing Center, ES)

Joint work of Eduardo Quinones, Sara Royuela, Maria A. Serrano

There is a visible trend in industry to adopt low-power hardware architectures with increasingly parallel execution capabilities supported by a variety of heterogeneous acceleration devices such as many-cores, and DSP fabrics, GPUs, SoC FPGAs. These architectures require the use of advanced parallel programming models for an efficient exploitation of their parallel capabilities. Parallel programming models however, cannot preclude the use of model-driven engineering, a very common practice in the software design as it facilitates the cyber/physical interaction description, abstracts the software/hardware architecture complexity and allows for early verification and validation performed on the models. This is the case of AUTOSAR, which is based on the concept of platform hardware independence, i.e., components are developed independently from their placement in the targeted processor architecture. Therefore, there is a need to bridge this gap by developing tools capable of generating parallel source code optimized for parallel heterogeneous platform, and run-time frameworks capable of respecting the guarantees devised at analysis time, while dynamically optimising the parallel execution to changing execution conditions.

3.21 Automotive System Design: Challenges of the Anthropocene

Sophie Quinton (INRIA – Grenoble, FR)

The current trend in the automotive industry towards automation and connectivity is raising a number of challenges in terms of cost-efficiency and dependability. At the same time, the conversation around the ongoing environmental crisis (in particular the climate emergency) makes it clear that we must either dramatically reduce our global resource consumption and pollution now, or face dire consequences in the near future. How do these two visions of the future relate?

The International Panel on Climate Change (IPCC) declared in 2018 that we must cut our CO2 emissions by 50% by 2030, reaching net zero around 2050, if we wanted to keep the global temperature in 2100 within 1.5°C over pre-industrial levels [1]. “Business as usual” scenarios (which we have followed since) lead to a temperature increase that could reach 5°C or even more by the end of the century[1]. In comparison, the average global temperature around 21,000 years ago (during the last glacial period) was 6°C colder than today's, and at the time a large part of Northern Europe was covered by several kilometers of ice [3, 4]. The impact on nature of such a huge change in a such a short time is hard to anticipate, and the situation is unprecedented.

Beyond climate change, the Earth is a complex system with many interdependent feedback loops [5]. For example, the devastating fires in Australia at the end of 2019, which have been caused by a severe drought, have released an estimated 250 million tons of CO2 in the atmosphere [6], which will in turn contribute to global warming. A number of planetary boundaries [5] have already been transgressed (e.g., those related to biodiversity and the cycle of nitrogen), meaning that non-linear, abrupt changes at the continental or planetary level cannot be excluded anymore (and this, independently from the direct impact of human activity) because tipping points may have been reached [7]. In addition, some changes are already unavoidable due to the latency in some natural phenomena such as the rising level of oceans [8]. Whether we choose to act or not, we will have to deal with increasingly degrading conditions, the scale of which partly depends on our actions.

In this context, the drivers for action at all levels of society (individuals, companies, etc.) are of different natures. First, the fact that parts of the world will become uninhabitable [9] provides a powerful ethical motivation. A second, more personal reason would be the understanding that climate change will impact the entire population, either directly (e.g. hydric stress and an increased frequency and intensity of extreme meteorological episodes [1]) and indirectly (massive migrations are expected[2]). Finally, a practical reason for companies to address the environmental crisis is the fact that there is a good chance that drastic regulations will come into place – and, given the situation, we should all welcome and prepare for such measures[3].

So, what role does the automotive industry play in this? If we focus on greenhouse gas (GHG) emissions, it appears that about 14% of GHG emissions were directly due to the transportation sector in 2010, a major part of it due to cars [13]. Besides, electricity/heat production itself represents about 25% of all emissions. This underlines that the challenge for the automotive industry cannot be reduced to the shift from fossil fuel propulsion to electric propulsion: electricity itself is not "clean". Whether it is possible to decarbonize energy production is an open problem because of the enormous amount of mineral resources we would need to extract for this (based on current technology for renewable energy production) [14].

Dramatically reducing the overall energy consumption of the automotive sector is therefore an urgent necessity. Autonomous driving is often seen as a potential enabler for this reduction, as it increases driving efficiency and could help reducing traffic congestion. For example, the following claim can be found on the website of Synopsys [15]: *"The real promise of autonomous cars is the potential for dramatically lowering CO2 emissions. In a recent study, experts identified three trends that, if adopted concurrently, would unleash the full potential of autonomous cars: vehicle automation, vehicle electrification, and ridesharing."*

Interestingly, the quoted study [16] also states that "Ride-sharing and renewable energy sources [are] critical to its success." Further [17]: "Our central finding is that while vehicle electrification and automation may produce potentially important benefits, without a corresponding shift toward shared mobility and greater use of transit and active transport, these

[1] More recent publications are even more pessimistic [2].
[2] The most widely accepted figure is 200 million climate migrants by 2050 [11].
[3] Interestingly, carmakers are among key opponents of climate action [12].

two revolutions could significantly increase congestion and urban sprawl, while also increasing the likelihood of missing climate change targets." The reason for that is the so-called rebound effect [18]: The energy benefits of increased driving efficiency offered by autonomous cars would be offset by the steep increase in their use because of the much lower time costs they enable (possibility to work in the car, no need to park, etc.). Such effects are substantial even for ridesharing where modal shifts from public transit and active modes to private cars, longer traveling distances, and increase of urban sprawl are also expected [19].

In conclusion, scenarios in which autonomous driving does not lead to an increase in global CO2 emissions are based on very strong assumptions. In other words, the risk of backfire (i.e., the introduction of autonomous driving actually increases CO2 emissions) is high. Moreover, the contribution of autonomous driving in successful scenarios appears to be limited compared to simple measures such as vehicle right-sizing or de-emphasized performance. Finally, the entire lifecycle of the vehicules (production, usage, end of life) should be covered, with issues arising related to batteries and reparability.

The one thing for which a consensus seems to exist is that we need to rethink the role of cars in our societies – that may be the largest challenge faced by the automotive industry.

Post-scriptum. There was a breakout session on "Cars and Climate Change" at the Dagstuhl seminar following this presentation. Ten people participated. Discussions centered around regulations coming into effect in various countries and their blind spots, e.g. regarding the impact of production and the need to tackle the rebound effect. Questions about the reparability of autonomous cars and a possibility of software obsolescence were also raised.

References

1. Special report of the Intergovernmental Panel on Climate Change (IPCC) on global warming of 1.5°C. https://www.ipcc.ch/sr15/
2. Two French Climate Models Consistently Predict a Pronounced Global Warming. CNRS Press release. September 2019.
3. https://en.wikipedia.org/wiki/Last_Glacial_Period
4. https://en.wikipedia.org/wiki/Quaternary_glaciation
5. W. Steffen et al. *Planetary Boundaries: Guiding Human Development on a Changing Planet*. Science Vol. 347, Issue 6223, 1259855.
6. Australia's Bushfires Have Emitted 250m Tonnes of CO2, Almost Half of Country's Annual Emissions. The Guardian, 13 December 2019.
7. T. M. Lenton, J. Rockström, O. Gaffney, S. Rahmstorf, K. Richardson, W. Steffen, and H. J. Schellnhuber. *Climate Tipping Points – Too Risky to Bet Against*. Nature 575, 592-595 (2019).
8. Special report of the IPCC on the ocean and cryosphere in a changing climate. https://www.ipcc.ch/srocc/
9. C. Mora et al. *Global Risk of Deadly Heat*. Nature Climate Change volume 7, pages 501-506 (2017).
10. *Why Does Europe Need to Limit Climate Change and Adapt to Its Impacts?* European Environment Agency, February 2020. https://www.eea.europa.eu/highlights/why-does-europe-need-to
11. Migration and Climate Change. International Organization for Migration (IOM). IOM Migration Research Series, vol. 31.
12. Carmakers Among Key Opponents of Climate Action. The Guardian, 10 October 2019.
13. https://www.epa.gov/ghgemissions/global-greenhouse-gas-emissions-data#Sector
14. O. Vidal, B. Goffé, and N. Arndt. *Metals for a Low-carbon Society*. Nature Geoscience volume 6, pages 894–896 (2013).

15 https://www.synopsys.com/automotive/what-is-autonomous-car.html
16 https://www.itdp.org/2017/05/03/3rs-in-urban-transport/
17 L. Fulton, J. Mason, and D. Meroux. *Three Revolutions in Urban Transportation.* ITDP and UC Davis research report.
18 M. Taiebat, S. Stolper and M. Xu. *Forecasting the Impact of Connected and Automated Vehicles on Energy Use: A Microeconomic Study of Induced Travel and Energy Rebound*. Applied Energy Volume 247, 1 August 2019, Pages 297–308. https://doi.org/10.1016/j.apenergy.2019.03.174
19 N. Coulombel, V. Boutueila, L. Liu , V. Viguié and B. Yind. *Substantial Rebound Effects in Urban Ridesharing: Simulating Travel Decisions in Paris, France*. Transportation Research Part D: Transport and Environment Volume 71, June 2019, Pages 110–126. https://www.sciencedirect.com/science/article/pii/S1361920918303201

3.22 AnyDSL: A Partial Evaluation Framework for Programming High-Performance Heterogeneous Systems

Roland Leißa (Universität des Saarlandes – Saarbrücken, DE)

Joint work of Roland Leißa, Klaas Boesche, Sebastian Hack, Arsène Pérard-Gayot, Richard Membarth, Philipp Slusallek, André Müller, Bertil Schmidt
Main reference Roland Leißa, Klaas Boesche, Sebastian Hack, Arsène Pérard-Gayot, Richard Membarth, Philipp Slusallek, André Müller, Bertil Schmidt: "AnyDSL: a partial evaluation framework for programming high-performance libraries", PACMPL, Vol. 2(OOPSLA), pp. 119:1–119:30, 2018.
URL http://dx.doi.org/10.1145/3276489

AnyDSL advocates programming high-performance code using partial evaluation. We present a clean-slate programming system with a simple, filter-based, online partial evaluator that operates on a CPS intermediate representation. Our system exposes code generation for accelerators (vectorization/parallelization for CPUs, GPUs, and FPGAs) via compiler-known higher-order functions that can be subjected to partial evaluation. This way, generic implementations can be instantiated with target-specific code at compile time.

In our experimental evaluation we present three extensive case studies from image processing, ray tracing, and genome sequence alignment. We demonstrate that using partial evaluation, we obtain high-performance implementations for CPUs and GPUs from *one* language and *one* code base in a *generic way.* The performance of our codes is mostly within 10%, often closer to the performance of multi man-year, industry-grade, manually-optimized expert codes that are considered to be among the top contenders in their fields.

3.23 DAPHNE – An Automotive Benchmark Suite for Parallel Programming Models on Embedded Heterogeneous Platforms

Lukas Sommer (TU Darmstadt, DE)

Joint work of Lukas Sommer, Florian Stock, Leonardo Solis-Vasquez, Andreas Koch
Main reference Lukas Sommer, Florian Stock, Leonardo Solis-Vasquez, Andreas Koch: "DAPHNE – An automotive benchmark suite for parallel programming models on embedded heterogeneous platforms: work-in-progress", in Proc. of the International Conference on Embedded Software Companion, New York, NY, USA, October 13-18, 2019, p. 4, ACM, 2019.
URL http://dx.doi.org/10.1145/3349568.3351547

Due to the ever-increasing computational demand of automotive applications, and in particular autonomous driving functionalities, the automotive industry and supply vendors are starting to adopt parallel and heterogeneous embedded platforms for their products.

However, C and C++, the currently dominating programming languages in this industry, do not provide sufficient mechanisms to target such platforms efficiently. Established parallel programming models such as OpenMP and OpenCL on the other hand are tailored towards HPC systems.

To assess the applicability of established parallel programming models to automotive workloads on heterogeneous platforms, we present the DAPHNE benchmark suite [1]. The suite contains typical automotive workloads extracted from the open-source framework Autoware and implementations for each kernel in OpenCL, CUDA and OpenMP. Next to details on the benchmark suite, we share lessons learned during the implementation in [2], also with regard to non-functional aspects such as programmer productivity and maintainability, and give an outlook on potentially interesting developments.

References

1 L. Sommer, F. Stock, L. Solis-Vasquez, and A. Koch. *Daphne – An Automotive Benchmark Suite for Parallel Programming Models on Embedded Heterogeneous Platforms: Work-in-progress* in Proceedings of the International Conference on Embedded Software Companion, ser. EMSOFT '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3349568.3351547.

2 L. Sommer, F. Stock, L. Solis-Vasquez, and A. Koch, *FAT-Schriftenreihe 317 – EPHoS: Evaluation of Programming-Models for Heterogeneous Systems*, Tech. Rep., 2019. [Online]. Available: https://www.vda.de/de/services/Publikationen/fat-schriftenreihe-317.html.

3.24 The Role of Synchronized Time for Safe Integration of Heterogeneous Software Applications

Wilfried Steiner (TTTech Computertechnik – Wien, AT)

The use of synchronized time to establish system-wide real-time and safety properties becomes more and more mainstream. Today, we find this basic idea in the form of "time-aware shaping" [1] as part of the IEEE 802.1 TSN standards, as "time-coordinated computing" as technology brand from a large semiconductor vendor, as well as the "time-triggered architecture" [2] in the academic literature. Indeed, synchronized time combined with offline task and message scheduling is also the core paradigm of the safe execution platform MotionWise that is deployed in automotive series production. In this talk I will review the core concepts of time synchronization and offline scheduling and give a quick overview of MotionWise. Furthermore, I will discuss current developments and potential research challenges in this domain.

References

1 W. Steiner, S. Craciunas, and R. Serna Oliver. *Traffic planning for time-sensitive communication.* IEEE Communications Standards Magazin, vol. 2, no. 2, pp. 42-47, 2018.

2 H. Kopetz and G. Bauer, *The Time-Triggered Architecture.* Proceedings of the IEEE, vol. 91, no. 1, pp. 112-126, 2003.

3.25 Taming Unpredictability: Leveraging Weakly-hard Constraints in Design and Adaptation

Qi Zhu (Northwestern University – Evanston, US)

Joint work of Qi Zhu, Rolf Ernst, Chung-Wei Lin, Wenchao Li, Samarjit Chakraborty, Hyoseung Kim
Main reference Chao Huang, Wenchao Li, Qi Zhu: "Formal verification of weakly-hard systems", in Proc. of the 22nd ACM International Conference on Hybrid Systems: Computation and Control, HSCC 2019, Montreal, QC, Canada, April 16-18, 2019, pp. 197–207, ACM, 2019.
URL http://dx.doi.org/10.1145/3302504.3311811
Main reference Hengyi Liang, Zhilu Wang, Debayan Roy, Soumyajit Dey, Samarjit Chakraborty, Qi Zhu: "Security-Driven Codesign with Weakly-Hard Constraints for Real-Time Embedded Systems", in Proc. of the 37th IEEE International Conference on Computer Design, ICCD 2019, Abu Dhabi, United Arab Emirates, November 17-20, 2019, pp. 217–226, IEEE, 2019.
URL http://dx.doi.org/10.1109/ICCD46524.2019.00035
Main reference Chao Huang, Kacper Wardega, Wenchao Li, Qi Zhu: "Exploring weakly-hard paradigm for networked systems", in Proc. of the Workshop on Design Automation for CPS and IoT, DESTION@CPSIoTWeek 2019, Montreal, QC, Canada, April 15, 2019, pp. 51–59, ACM, 2019.
URL http://dx.doi.org/10.1145/3313151.3313165
Main reference Hyunjong Choi, Hyoseung Kim, Qi Zhu: "Job-Class-Level Fixed Priority Scheduling of Weakly-Hard Real-Time Systems", in Proc. of the 25th IEEE Real-Time and Embedded Technology and Applications Symposium, RTAS 2019, Montreal, QC, Canada, April 16-18, 2019, pp. 241–253, IEEE, 2019.
URL http://dx.doi.org/10.1109/RTAS.2019.00028

In automotive systems, limited resources and hard timing constraints often make it difficult to apply design changes for adding new functionality or correcting existing ones, and to adapt the systems under changing and challenging environment. In this talk, I will present some of our initial thoughts in leveraging weakly-hard timing constraints, where operations are allowed to occasionally miss deadlines in a bounded manner, to improve systems' capability in accommodating changes at design-time and runtime. I will introduce our preliminary works in 1) formally verifying the safety of control functions under weakly-hard constraints at the functional layer, 2) exploring the addition of security monitoring tasks under given weakly-hard constraints at the software layer, and 3) setting weakly-hard constraints cross functional and software layers to facilitate system retrofitting and adaptation.

3.26 Breaking Automotive Traditions

Dirk Ziegenbein (Robert Bosch GmbH – Stuttgart, DE)

Joint work of Selma Saidi, Sebastian Steinhorst, Arne Hamann, Dirk Ziegenbein
Main reference Selma Saidi, Sebastian Steinhorst, Arne Hamann, Dirk Ziegenbein, Marko Wolf: "Future automotive systems design: research challenges and opportunities: special session", in Proc. of the International Conference on Hardware/Software Codesign and System Synthesis, CODES+ISSS 2018, part of ESWEEK 2018, Torino, Italy, September 30 – October 5, 2018, p. 2, IEEE / ACM, 2018.
URL http://dx.doi.org/10.1109/CODESISSS.2018.8525873

The landscape in the automotive industry is rapidly changing, driven by new automotive functionalities and applications such as autonomous driving and fueled by the entry of new players from the IT industry. Consequently, we witness the emergence of new trends in the automotive field imposing new challenges at the hardware and software level and at the way future automotive systems will be designed and developed. This includes the integration of formerly separated function domains onto centralized computing platforms, leading to a heterogeneous mix of applications with different models of computation, and the diversification and specialization of hardware platforms (comprising e.g., application cores, safety cores, deep learning accelerators, issued across several control units and connected by networks). Such platforms are required to meet the tremendous increase of computing power enforced by functionalities such as automated driving or on-line optimization of operating strategies for electrified powertrains. Furthermore, there is an increase of the connectivity beyond vehicle boundaries, such as vehicle-to-vehicle and vehicle-to-infrastructure communication (V2X) leading to systems-of-systems integration problems including function deployment from control unit over edge to cloud infrastructure as well as questions of how to guarantee sufficient availability of communication links and security on these connections.

These new trends constitute a real paradigm shift in the way classical automotive systems are viewed, thereby heavily challenging established methods and methodologies in automotive systems design. This provides a great opportunity for practitioners from both academia and industry to rethink the design of future automotive systems in order to accommodate high efficiency, safety and security requirements.

This talk first discusses the main trends that are currently driving innovation in the automotive industry. Then an overview of the emerging challenges and application constraints necessary to comply with the increased requirements with respect to cost, variability, support of legacy software and dependability is given. The goal is to raise awareness about emerging needs and application constraints in automotive.

4 Working groups

4.1 Modeling Hardware Dependencies and Software Dependencies

Jerónimo Castrillón-Mazo (TU Dresden, DE), Lulu Chan (NXP Semiconductors – Eindhoven, NL), Oliver Kopp (Mercedes Benz AG – Stuttgart, DE), Roland Leißa (Universität des Saarlandes – Saarbrücken, DE), Albrecht Mayer (Infineon Technologies – München, DE), Philipp Mundhenk (Autonomous Intelligent Driving GmbH – München, DE), Philipp Obergfell (BMW AG – München, DE), and Eduardo Quinones (Barcelona Supercomputing Center, ES)

4.1.1 Introduction

This discussion group covered the topics of capturing components and their dependencies. For instance, software A may run on electronic control unit (ECU) E only, whereas software B "just" requires CPU architecture M and operating system O [1].

We saw following aspects in the context of dependencies:

- Dependencies between software components
- Dependencies from software to hardware (platforms), for instance "high performance"/ accelerators, CPU, GPU, timing constraints, i.e., requirements of software to hardware
- Describing hardware capabilities (both in terms of dependencies as required by software, or as assignable resource for allocation)
- Type system of described data
- Non-functional requirements of software & hardware components and communicated data
- Deployment of software to hardware components

We identified function developers, application developers, and system integrators as roles.

- A function developer delivers functionalities of ECUs.
- An application developer delivers applications usable by customers.
- An application developer uses functions delivered by function developers.
- A system integrator combines functionality and application into a system.

We identified the following scenarios involving the aspects of HW/SW dependencies and the roles described above:

- S1: Developers need to describe their required environment.
- S2: Developers need to be aware of the provided environment. It may be the case that developers are unaware of certain properties of their environment. These properties need to be made explicit.
- S3: A system integrator can use information such as timing constraints or timing boundaries for developing the final distribution of software. In some instances, domain-specific knowledge is required for an optimal placement of functionalities. For example, the assignment of accelerated code is not straight-forward.
- S4: Leveraging the information available during runtime to, (for instance, triggering adoption of configuration, movement of execution, redeployment of functionalities)
- S5: Deduction of execution times based on the function in software, its mapping to hardware, and available/used resources
- S6: Configure system from a high-level API, e.g.:
  - a) Middleware configures underlying TSN-layer (based on offered options).
  - b) Migrate from ECU to HPU (accelerators)

4.1.2 Possible Solutions

We briefly discussed possible solutions to capture the HW/SW dependencies:

- High-level programming models, where an abstract model of and estimations (in terms of latency and energy consumption) for the application can be passed to the runtime. This can enable model-based runtime decision making according to available resources and requirements (cf. [5, 6]).
- Architecture description languages such as UML (cf. [4]).
- Languages supporting the description of composite cloud applications. In that field, dependencies and resource management are also a discussed topic ([3, 2]).
- With respect to the hardware description for software tools the IEEE standard 2804-2019[7] is a promising standard.

Potentially, a single solution is insufficient, if a large number of developers and engineers from different backgrounds are to use the system to work together (e.g., embedded SW development vs. high-performance SW development vs. system integration). Different users might have different requirements in terms of representation of data. Translations between formats or different representations would be required.

4.1.3 Next Steps

In the group discussion, we identified the following next steps: First, we are going to use the above scenarios to deduct concrete requirements. Second, we craft end-to-end application scenarios to enable evaluations.

References

1 M. Staron, *Automotive Software Architectures – An Introduction*. Springer Internantional Publishing AG, doi:10.1007/978-3-319-58610-6, 2017.

2 U. Breitenbücher, C. Endres, K. Képes, O. Kopp, F. Leymann, S. Wagner, J. Wettinger, and M. Zimmermann. *The OpenTOSCA Ecosystem – Concepts & Tools*. European Space project on Smart Systems, Big Data, Future Internet – Towards Serving the Grand Societal Challenges, Volume 1: EPS Rome, SciTePress, 2016.

3 OASIS. *Topology and Orchestration Specification for Cloud Applications (TOSCA)*. https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tosca, 2020.

4 P. Obergfell, S. Kugele, and E. Sax. *Model-Based Resource Analysis and Synthesis of Service-Oriented Automotive Software Architectures*, 22nd International Conference on Model Driven Engineering Languages and Systems, IEEE/ACM, 2019.

5 A. Goens, R. Khasanov, M. Hähnel, T. Smejkal, H. Härtig, and J. Castrillon. *TETRiS: a Multi-Application Run-Time System for Predictable Execution of Static Mappings*. Proceedings of the 20th International Workshop on Software and Compilers for Embedded Systems, ACM, doi:10.1145/3078659.3078663, 2017.

6 R. Khasanov and J. Castrillon. *Energy-efficient Runtime Resource Management for Adaptable Multi-application Mapping*. Proceedings of the 2020 Design, Automation and Test in Europe Conference, Grenoble, 2020.

7 IEEE. *2804-2019 – IEEE Standard for Software-Hardware Interface for Multi-Many-Core*. 2020.

4.2 Weakly Hard Real-Time Models

Martina Maggio (Lund University, SE), Jerónimo Castrillón-Mazo (TU Dresden, DE), Lulu Chan (NXP Semiconductors – Eindhoven, NL), Rolf Ernst (TU Braunschweig, DE), Chung-Wei Lin (National Taiwan University – Taipei, TW), Zhu Qi (Northwestern University – Evanston, US), Eduardo Quinones (Barcelona Supercomputing Center, ES), Sophie Quinton (INRIA – Grenoble, FR), and Selma Saidi (TU Dortmund, DE)

We discussed the weakly hard model, its application domains, and what can be achieved using it. We concluded that the model itself is useful when, for example, safety-critical functions are executed on systems that are resource-constrained and evolve over time. For example, when an update or upgrade happens, functionalities are added and this can disrupt the behavior of the real-time systems as it was designed in the first place. Another important use of the model is handling faults and identifying risky situations. A system that can withstand two deadline misses in a window of four could be put in an alarm state after one deadline miss occurs, monitors can be activated, and countermeasures can be taken to ensure the correct behavior despite misses.

Application domains that were identified include: (1) security attacks, (2) control tasks and robotics, (3) image transmission. For security attacks, we talked about protocols in which a given sequence of messages has to be delivered and disrupting the transmission of some of these messages could lead to damage and problems. One of the main application domains would be vehicle to vehicle communication with wireless communication. For control tasks, there is a lot of research in trying to connect weakly hard guarantees and functional properties (e.g., despite missing deadlines, the control system is still stable and has good performance in terms of speed of convergence to the objective and integral of the squared error). We discussed if this is the right model to use for this case, without any conclusive result. It seems that the weakly hard model is a good match for performance handling, but less good for stability analysis, i.e., the analysis is mathematically involved and it is hard to make it scale and generalize it. For image transmission, we discussed cases in which images composed of multiple pixels have to be transmitted in a way such that the content is recognizable despite some pixels not being transmitted correctly.

Generally speaking, these different application models would have very different m-K parameters. For control systems, m and K would be close to one another, while for image transmission m and K would be quite far from each other (an image has very many pixels, thus a large K, and only a few of them can be erroneous, thus m is small). We then briefly discussed how to obtain these parameters. One example is using truncated probability distributions and estimating the parameters from previous runs of the applications. We noted that it might be quite difficult to derive the parameters and also to analyze the corresponding models.

Another point that we addressed is what to do with these models. Two ideas emerged: the first one is to understand when it is important to switch to some safety mode; while the second one is to determine how to handle slack time reclamation (if two tasks can be given some additional time to execute, maybe it is better to prioritize based on how many deadlines can they miss in their future executions based on their window sizes, rather than looking at one single instance of the task execution).

4.3 Machine Learning in Cyber-Physical Systems

Frank Mueller (North Carolina State University – Raleigh, US), Bart Besselink (University of Groningen, NL), Alessandro Biondi (Sant'Anna School of Advanced Studies – Pisa, IT), Lulu Chan (NXP Semiconductors – Eindhoven, NL), Jyotirmoy Deshmukh (USC – Los Angeles, US), Masaki Gondo (eSOL – Tokyo, JP), Baik Hoh (Toyota Motors North America – Mountain View, US), Peter Gorm Larsen (Aarhus University, DK), Mark Lawford (McMaster University – Hamilton, CA), Martina Maggio (Lund University, SE), Albrecht Mayer (Infineon Technologies – München, DE), Zhu Qi (Northwestern University – Evanston, US), Lukas Sommer (TU Darmstadt, DE), Wilfried Steiner (TTTech Computertechnik – Wien, AT), and Seyhan Uçar (Toyota Motors North America – Mountain View, US)

This is the recording of a working group on machine learning for cyber-physical systems, in which more people than recorded participated.

We started the discussion with a challenging example. Suppose that a machine learning system, used for vision and object recognition in an autonomous car, is not able to detect a stop sign. Using some ground truth information (e.g., information received with other sensors), the stop sign is identified, and the learning system is able to rewind and go back to the frame where the stop sign could have been detected. This enables for continuous on-vehicle learning. We asked ourselves how is this possible to achieve. Some aspect of this problem involve the cloud and the ability to "learn from mistakes", autonomously, and possibly cooperatively (with other vehicles).

Generally speaking we moved the discussion on the topics of how it is possible to detect and recognize bad situations, and how it is possible to avoid overfitting in these cases. We discussed the need for going back to supervised learning for these cases and what the update of the learning procedures will mean. However, we noted many open problems, like the presence of dynamically moving objects. We need to analyze the root cause for the error (the missed detection) and to understand how to correct it on the fly.

We then started talking about uncertainty and the guarantees that can be provided either with machine learning or with other techniques, linking this to model order reduction to reduce the complexity of the models that are used for control (e.g., trajectory planning). We had a brief discussion on the relation between optimal control and reinforcement learning. We noted that a recently published paper by Benjamin Recht tries to link the two [1].

References

1 B. Recht. *A Tour of Reinforcement Learning: The View from Continuous Control*. Annual Review of Control, Robotics, and Autonomous Systems, 2(1): 253–279, 2019

4.4 HW/SW Architecture Exchange

Philipp Mundhenk (Autonomous Intelligent Driving GmbH – München, DE)

Joint work of all participants of Dagstuhl Seminar 19502

The breakout session was set up as an interactive session to identify upcoming trends in automotive architectures and to discuss which of the challenges are already being addressed by which solutions. The majority of participants of Dagstuhl Seminar 19502 took part in this session.

The focus of the session was interactively set on determinism for new architectures. Ensuring such determinism is non-trivial with the emergence of new technologies in safety-critical systems, such as Ethernet-based, service-oriented communication, microprocessors, and additional abstraction layers in software, adding complexity [2].

4.4.1 Background

New software platforms add additional layers, supporting developers, simplifying application development and increasing speed of innovation [1]. However, each layer being introduced into a system adds complexity and variability. Ensuring correct and deterministic behavior in such a system is not trivial. For example, scheduling problems arise, e.g., spatial or time-based scheduling of computation on hardware accelerators. Often, this is made worse due to the proprietary nature of many of the hardware components, especially accelerators, such as GPUs. Typically, not much information is known about these, making achieving determinism more difficult. Another example is the mapping of scheduling algorithms and schedulable units on the layer of software platform, operating system, hypervisor, device, etc.

4.4.2 Approaches

Due to the large nature of the problem there is no single solution. Different approaches need to be combined. The following approaches, addressing parts of the problem, have been discussed in this breakout session:

- Limiting every layer in its potentially unlimited freedom could help reduce the overall uncertainty in the system. E.g., timing restrictions can be imposed on the hypervisor, the operating system, etc., and not just on the level of application. The same goes for memory bounds, etc.
- Design time checks will be required wherever possible. The developers and integrators should not be allowed to perform certain actions, if these are already known to be breaking the system. E.g., logically connecting an ASIL-D input of one component to an ASIL-A output of another component can already be detected at design time, even if the system uses service-oriented communication and sets up connections at runtime only.
- The behavior defined at design time needs to be enforced at runtime. E.g., through configuration of the operating system, or the software platform. Furthermore, the correct behavior needs to be monitored at runtime, such that failure reactions can be triggered when abnormal behavior, compared to the design time specification, is detected.
- The execution behavior of microprocessors today is inherently nondeterministic. Due to the use of multi-level caches, virtual memory, etc. it is not reasonably possible to achieve deterministic behavior on microprocessors. This might, however, not hold forever. Chip

designers are working on making their processors more predictable. This would be a long-term approach and not an immediate solution.

- Another layer of protection can be achieved through introduction of a parallel computation path with plausibility computations. These parallel paths do not need to perform the full set of computations, but can be more lightweight, e.g., computing boundaries, and can be implemented on a more deterministic system, such as a microcontroller.
- Such a reliable path, if computation is accurate enough, could also be used as a fallback path, in case the main path fails during operation.
- In case a redundant, deterministic path has less computational performance, the environment of the autonomous system can also be adapted. E.g., the speed of a vehicle could be reduced, allowing longer reaction times, thus allowing computation to be slower and less powerful computation units to be sufficient.

4.4.3 Open Challenges

Despite the numerous approaches discussed in this session, a number of open challenges have been identified, e.g.,:

- Is achieving reliability of the complete system, including microprocessor, GPU, hardware accelerators, operating system, middleware, etc. a realistic goal in the long-term? There is no definitive answer to this question, yet.
- How exactly can redundant paths be designed? Is a reliable path always required? Or could e.g., an unreliable path, plus reliable plausibility checkers be sufficient? Could multiple unreliable paths and reliable voters be used? Are there other approaches?
- When using design time checks, how is illicit behavior defined? How can it be ensured that the complete set of illicit behavior is covered? How can this be integrated into the many different approaches (e.g., development frameworks, continuous integration systems, etc.) at design time? How to enforce this correct behavior at runtime in an efficient and reliable manner?
- When using partial checks in the individual layers of the system, how can these be enforced? What are reasonable checks and bounds (e.g., for the number of obstacles detected)? How can the boundaries be monitored in an efficient manner?

- When adapting to the environment, e.g., through slowing down the vehicle, it needs to be ensured that the adjustment is sufficient. Depending on the available reliable compute performance and the complexity of required computation, it might e.g., still not be sufficient to slow down the vehicle, even to walking speed.
- How to deal with input to the system that is inherently indeterministic? The interpretation of sensor inputs e.g., will likely always introduce indeterminism to the system. How can such input, which is essential to the functionality of the system, be processed in a deterministic system?

4.4.4 Conclusion

This session identified a large number of challenges to and a number of solutions for dependability in the area of highly complex mixed-criticality systems on heterogeneous architectures. A large amount of research is required on all aspects of the system, from hardware over different software layers to processes and legislation in order to address these challenges.

References

1 P. Mundhenk, E. Parodi, and R. Schabenberger. *Fusion: A Safe and Secure Software Platform for Autonomous Driving.* In Proceedings of the 2nd Workshop on Autonomous Systems Design (ASD 2020). Grenoble, France, 2020.

2 L. Bauer, M. Damschen, D. Ziegenbein, A. Hamann, A. Biondi, G. Buttazzo, and J. Henkel. *Analyses and Architectures for Mixed-critical Systems: Industry Trends and Research Perspective.* In Proceedings of the International Conference on Embedded Software Companion (EMSOFT '19). New York, USA, Article 13, 1–2.

4.5 On the Relation between Programming Models, Computational/Execution Models and Software Platforms in Automotive

Selma Saidi (TU Dortmund, DE), Alessandro Biondi (Sant'Anna School of Advanced Studies – Pisa, IT), Rolf Ernst (TU Braunschweig, DE), Sabine Glesner (TU Berlin, DE), Oliver Kopp (Mercedes Benz AG – Stuttgart, DE), Roland Leißa (Universität des Saarlandes – Saarbrücken, DE), Philipp Mundhenk (Autonomous Intelligent Driving GmbH – München, DE), Philipp Obergfell (BMW AG – München, DE), Eduardo Quinones (Barcelona Supercomputing Center, ES), Lukas Sommer (TU Darmstadt, DE), and Dirk Ziegenbein (Robert Bosch GmbH – Stuttgart, DE)

The breakout session discussed the relationship between (parallel) programming models, software platforms and models of execution.

Software platforms in automotive integrate multiple software components including applications, middleware, operating systems, from different software providers and are executed on a heterogeneous collection of hardware including microcontrollers, CPUs and GPUs. Pthread is often used as an execution model (e.g., in AUTOSAR Adaptive) to control multiple workflows that overlap over time.

With the adoption of high-performance platforms in automotive, parallel programming models are becoming prevalent, targeting heterogenous hardware platforms composed often of a combination of CPUs and GPUs. Several programming models like CUDA and OpenMP, implement different models of execution (i.e., execution semantics) in terms of data communication (e.g. shared memory or message passing) and synchronization (e.g., implicit or explicit). Unlike HPC systems, automotive systems have different/additional requirements in terms of heterogeneity, timing predictability, data versioning, etc. Therefore, a computational model (can also be referred to as System Execution Model) is further needed to coordinate the execution of different software components. The role of computational models is to guarantee the integration of different software layers and orchestrate their "correct" execution. Correct execution requires predictable or defined execution and communication timings, data consistency, data versioning, etc. The Logical Execution Time (LET) Model is an example of computational models. Therefore, the programming model is different from the computational model and this latter is often much more complicated as it must as well guarantee non-functional properties and a seamless integration between software layers.

(During the discussion at Dagstuhl, the term execution model was interchangeably used to designate programming models, execution semantics and system-level computational execution models. However, a clear distinction between the two is needed).

It remains however unclear how programming models, used by software programmers to implement a given function, interface with computational models, used by system integrators to guarantee a system-level correct execution. Physical timing (e.g. period, timing constraints, end-to-end latencies) for instance define properties of the software and should be part of the programming model. These properties are later further propagated from the programming model to the computational model responsible for their implementation by structuring the execution and synchronization in the software architecture. It seems that a clear unified software platform view integrating both programming models and computational models is missing. How can the programming model and computational model be integrated in the automotive software stack? Regarding hardware resources allocation, are all hardware resources allocated by the computational model? What's the role/requirements derived from the programming model regarding tasks partitioning, resources allocation and scheduling? Which level of abstraction should be used for scheduling, communication and synchronization? These are all important open questions to be addressed.

4.6 Benchmarking Efforts for Future HW/SW Platforms

Lukas Sommer (TU Darmstadt, DE), Bart Besselink (University of Groningen, NL), Alessandro Biondi (Sant'Anna School of Advanced Studies – Pisa, IT), Jerónimo Castrillón-Mazo (TU Dresden, DE), Lulu Chan (NXP Semiconductors – Eindhoven, NL), Wanli Chang (University of York, GB), Thidapat Chantem (Virginia Polytechnic Institute – Arlington, US), Rolf Ernst (TU Braunschweig, DE), Oliver Kopp (Mercedes Benz AG – Stuttgart, DE), Mark Lawford (McMaster University – Hamilton, CA), Roland Leißa (Universität des Saarlandes – Saarbrücken, DE), Martina Maggio (Lund University, SE), Philipp Mundhenk (Autonomous Intelligent Driving GmbH – München, DE), Philipp Obergfell (BMW AG – München, DE), Eduardo Quinones (Barcelona Supercomputing Center, ES), Selma Saidi (TU Dortmund, DE), Lea Schönberger (TU Dortmund, DE), and Dirk Ziegenbein (Robert Bosch GmbH – Stuttgart, DE)

The closing session of the seminar also saw a discussion about benchmarking efforts for future automotive HW/SW platforms. The participants agreed that the availability of representative benchmarks is critical for the development of central components, such as programming models, compilers, runtime systems and the evaluation of hardware platforms. Therefore, academia and industry should collaborate to provide open benchmarks for use in research and development.

One such approach is the open-source DAPHNE benchmark suite [4], which was created with the intent to provide a number of representative automotive workloads under the permissive Apache v2 license. The suite contains typical automotive workloads extracted from the open-source framework Autoware and implementations for each kernel in OpenCL, CUDA and OpenMP.

In order to extend this open benchmark suite, the original authors of the benchmark seek for contributions to the benchmark. The form of the contribution can be manifold, for example:

- Hints to relevant algorithms used in real-world automotive applications to be implemented as benchmark kernel.
- Implementations of automotive algorithms in any programming language to be added to the benchmark and ported to the different programming models.
- Representative data-sets, e.g. as captured by sensors and cameras, for use with the benchmark algorithms.

We invite everyone to contribute to this open benchmarking approach through the Github-page of the DAPHNE project [5] or via e-mail to sommer@esa.tu-darmstadt.de

Another source for benchmarks is the community forum of the WATERS workshop [1] where the models of the WATERS industrial challenges are stored. The purpose of these challenge is to share ideas, experiences and solutions to concrete timing verification problems issued from real industrial case studies. It also aims at promoting discussions, closer interactions, cross fertilization of ideas and synergies across the breadth of the real-time research community as well as industrial practitioners from different domains. Here the focus is on non-functional performance models of real-world systems, ranging from engine control [2] to automated driving [3].

References

1 WATERS @ ECRTS Community Forum. *Industrial Challenge Models*. https://www.ecrts.org/forum/ .

2 S. Kramer, D. Ziegenbein, and A. Hamann. *Real World Automotive Benchmarks For Free*. WATERS 2019, Lund, 2019, https://www.ecrts.org/forum/download/file.php?id=9&sid=d5f57b4d87f08f919fd0a366b31bab49.

3 F. Rehm, D. Dasari et al. *System Performance Modelling of Heterogeneous HW Platforms: An Automated Driving Case Study*. 22nd Euromicro Conference on Digital System Design (DSD), Dubrovnik, 2019. https://doi.org/10.1109/DSD.2019.00060.

4 L. Sommer, F. Stock, L. Solis-Vasquez, and A. Koch, "Daphne – An Automotive Benchmark Suite for Parallel Programming Models on Embedded Heterogeneous Platforms: Work-in-progress," in *Proceedings of the International Conference on Embedded Software Companion*, ser. EMSOFT '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3349568.3351547.

5 L. Sommer, F. Stock, L. Solis-Vasquez, and A. Koch, "Darmstadt Automotive Parallel HeterogeNEous (DAPHNE) Benchmark-Suite," https://github.com/esa-tu-darmstadt/daphne-benchmark, 2019.

4.7 Modularizing Control Systems

Dirk Ziegenbein (Robert Bosch GmbH – Stuttgart, DE), Bart Besselink (University of Groningen, NL), Peter Gorm Larsen (Aarhus University, DK), and Mark Lawford (McMaster University – Hamilton, CA)

Control systems are defined by their input-output behavior and their end-to-end cause and effect chains. A typical way of decomposing control software is however according to functional subsystems, which is also reflected in the predominant modeling style which typically features hierarchical dataflow and control flow blocks as, e.g., in Simulink. When maintaining and extending such models, changes or additional features usually require modifications to more than one of these subsystems and thus the respective software modules need to be changed in order to implement them. These modifications of multiple subsystems to handle different cases often results in a multitude of distributed "balconies" (nested if-then-else structures) often with state feedback through unit delays effectively resulting in distributed implicit state machines which are hard to understand and maintain.

In this breakout session, the participants discussed whether there are better ways to structure control systems in order to improve encapsulation and information hiding principles from classical software engineering [6] in control systems models. Finding such modularization approaches are of great importance considering the current automotive trends of increasing integration complexity as well as the update of vehicle functions after the vehicle's start of production (feature-over-the-air-update). Two approaches were discussed in more detail.

Simulink is widely used to develop control systems software in the automotive domain but the engineers using it often focus solely on the dataflow aspects of the problem which has been shown in general software engineering to result in systems with poor modularization [6]. While Simulink constructs such subsystems, libraries and model references can all be used to help structure Simulink models, all of these constructs have limitations when trying to perform system modularization based upon Parnas' information hiding principles [3]. Recently Mathworks has added a new language construct, Simulink functions, which can be used to help implement modular Simulink designs on an information hiding basis similar to modules in the C programming language [3]. Further study is needed to see if structuring Simulink models using the proposed information hiding principles leads to designs that are easier to maintain and comprehensible by the control domain engineers.

At Bosch, a method called SCODE (System Co-Design) Essential Analysis [2] has been developed which decomposes a control system in terms of discrete situations in the system context, the so-called functional invariants or modes. In these functional invariants the control system shows a coherent input-output behavior. The SCODE Essential Analysis is based on the Essential Systems Analysis [4], developed by McMenamin and Palmer originally for IT systems, and extends and modifies it to enable application for physically dominated systems.

The analysis follows a three-step-approach. In the first step, the problem space of the control system is defined by a Zwicky-Box in terms of input and output dimensions (aspects of the system or its context that cause or represent different system behaviors) and alternatives (possible values or value ranges of a dimension). In the second step, the problem space is partitioned into valid and invalid input and output sets, and modes are defined by assigning input sets to output sets. In the third step, the mode transitions are defined, i.e. it is specified which context changes cause a transition between system modes. The analysis is supported by a tool [5], which encodes the problem space as binary decision diagrams and supports the developer by guiding the analysis (e.g., to discover not yet covered parts of the problem space) and to guarantee properties of the system design (e.g. consistency and completeness of mode specifications as well as uniqueness and liveliness of mode transitions). Based on the analysis results, mode switching logic (in, e.g., Simulink, ASCET or C), test cases as well as documentation and reports can be generated automatically.

In related work, [1] attempts to simplify mode switching logic implemented in Simulink truth tables by identifying common behaviors and merging the resulting modes. Still within the Simulink setting, [7] discusses how Simulink models implicitly encoding state machines can be converted into StateFlow models effectively making the mode switching logic in data flow models explicit.

Given the above related works it seems that mode switching logic oriented functions are primary candidates for refactoring and alternative modularizations in order to improve control software designs.

References

1 M. Bialy, M. Lawford, V. Pantelic, and A. Wassyng. *A methodology for the simplification of tabular designs in model-based development.* In 2015 IEEE/ACM 3rd FME Workshop on Formal Methods in Software Engineering, pp. 47-53. IEEE, 2015.

2 M. Bitzer, M. Herrmann, and E. Mayer-John. *System Co-Design (SCODE): Methodology for the Analysis of Hybrid Systems.* at-Automatisierungstechnik, 67(9), 2019, https://doi.org/10.1515/auto-2019-0003.

3 M. Jaskolka. *A Comparison of Componentization Constructs for Supporting Modularity in Simulink*, SAE WCX 2020 World Congress Experience, April 2020.

4 S. McMenamin and J. Palmer. *Essential Systems Analysis.* Prentice-Hall, 1984, https://archive.org/details/essentialsystems00mcme.

5 ETAS GmbH. *SCODE Essential Analysis – Describe and Verify Complex Control Systems.* Whitepaper, 2016, https://www.etas.com/en/downloadcenter/24029.php.

6 Da. L. Parnas. *On the Criteria To Be Used in Decomposing Systems into Modules* (PDF). Communications of the ACM. **15** (12): 1053-1058, 1972, doi:10.1145/361598.361623.

7 S. Wynn-Williams, Z. Diskin, V. Pantelic, M. Lawford, G. Selim, C. Milo, M. Diab, and F. Weslati. *SL2SF: Refactoring Simulink to Stateflow.* In International Conference on Fundamental Approaches to Software Engineering, pp. 264-281. Springer, Cham, 2019.

4.8 SW Lifecycle of Dependable Evolving Systems

Dirk Ziegenbein (Robert Bosch GmbH – Stuttgart, DE), Rolf Ernst (TU Braunschweig, DE), Moritz Neukirchner (Elektrobit Automotive – Erlangen, DE), and Selma Saidi (TU Dortmund, DE)

Traditionally, automotive software was developed, tested and shipped bundled together with the hardware as an electronic control unit to the customer. Updates due to planned maintenance or bug removal were rather an exception and were performed by (re-)flashing the complete electronic control unit in the repair shop. Due to the increasing over-the-air connectivity beyond vehicle boundaries which has shown to create a new security attack surface which mandates fast fixing of security issues as well as to emerging business models of selling feature updates during the vehicle lifecycle, the frequency of such updates will increase significantly in the foreseeable future. Furthermore, due to the integration of several functionalities on so-called vehicle integration platforms, the updates will also become modular, exchanging or adding only parts of the software at a time. These two trends and the challenges they are causing were discussed in this breakout session.

With the increased frequency of updates, the whole release process will need to be highly automated. Note that the release process comprises not just the test and verification but also the fulfillment of all legal requirements and norms, like homologation or the proof of functional safety. While the test automation is already well on its way (even though the automated regression testing of cyber-physical systems still has a lot of challenges), the latter procedures are still dominated by a lot of manual work. Here, being able to achieve modularity, in the sense, that local changes in the system will also only have a clearly identifiable local impact on e.g. the safety case, is seen as a huge advantage over the current state-of-practice. In fact, the automotive standard for functional safety ISO 26262 is very inconclusive about requirements for maintenance such that more work is needed here.

The fact that maintenance will have to be offered over the whole vehicle life cycle poses additional challenges. The typical support and production cycles for microprocessors is much lower than the vehicle life cycle. Thus, one has to assume that during the end of the vehicle life cycle, the software might even need to be ported to new hardware generations as part of the maintenance process (a prominent example is Tesla, which is already offering upgrades to new vehicle computer generations to its customers). Similarly, the rapid development of automotive software platforms such as AUTOSAR Adaptive will lead to questions of whether keeping a system on its original SW platform version or continuously updating the platform version. Both strategies have their specific advantages and disadvantages. While keeping the platform version stable limits changes and reduces risks, it also requires the SW platform vendor to fix issues in all versions that are in the field. This long-term maintenance needs to be considered in the business model right from project start as it tremendously influences the cost of updates. Here, the automotive industry might look e.g. at consumer electronics and the update/maintenance models of SW platforms such as Android or iOS.

The participants of this breakout session agreed that this SW lifcycle topic for dependable evolving systems should be investigated in more detail. The organization of a special session at a top embedded systems conference is envisioned.

Participants

- Bart Besselink
University of Groningen, NL
- Alessandro Biondi
Sant'Anna School of Advanced Studies – Pisa, IT
- Jerónimo Castrillón-Mazo
TU Dresden, DE
- Lulu Chan
NXP Semiconductors – Eindhoven, NL
- Wanli Chang
University of York, GB
- Thidapat Chantem
Virginia Polytechnic Institute – Arlington, US
- Jyotirmoy Deshmukh
USC – Los Angeles, US
- Rolf Ernst
TU Braunschweig, DE
- Sabine Glesner
TU Berlin, DE
- Masaki Gondo
eSOL – Tokyo, JP
- Baik Hoh
Toyota Motors North America – Mountain View, US
- Oliver Kopp
Mercedes Benz AG – Stuttgart, DE
- Peter Gorm Larsen
Aarhus University, DK
- Mark Lawford
McMaster University – Hamilton, CA
- Roland Leißa
Universität des Saarlandes – Saarbrücken, DE
- Chung-Wei Lin
National Taiwan University – Taipei, TW
- Martina Maggio
Lund University, SE
- Albrecht Mayer
Infineon Technologies – München, DE
- Frank Mueller
North Carolina State University – Raleigh, US
- Philipp Mundhenk
Autonomous Intelligent Driving GmbH – München, DE
- Alessandra Nardi
Cadence – San Jose, US
- Moritz Neukirchner
Elektrobit Automotive – Erlangen, DE
- Philipp Obergfell
BMW AG – München, DE
- Maximilian Odendahl
Silexica – Köln, DE
- Zhu Qi
Northwestern University – Evanston, US
- Eduardo Quinones
Barcelona Supercomputing Center, ES
- Sophie Quinton
INRIA – Grenoble, FR
- Selma Saidi
TU Dortmund, DE
- Ignacio Sañudo Olmedo
University of Modena, IT
- Lea Schönberger
TU Dortmund, DE
- Lukas Sommer
TU Darmstadt, DE
- Wilfried Steiner
TTTech Computertechnik – Wien, AT
- Seyhan Uçar
Toyota Motors North America – Mountain View, US
- Dirk Ziegenbein
Robert Bosch GmbH – Stuttgart, DE

Report from Dagstuhl Seminar 19511

Artificial and Computational Intelligence in Games: Revolutions in Computational Game AI

Edited by

Jialin Liu[1], Tom Schaul[2], Pieter Spronck[3], and Julian Togelius[4]

1 **Southern Univ. of Science and Technology – Shenzen, CN,** `liujl@sustech.edu.cn`
2 **Google DeepMind – London, GB,** `schaul@google.com`
3 **Tilburg University, NL,** `p.spronck@gmail.com`
4 **New York University, US,** `julian@togelius.com`

Abstract

The 2016 success of Google DeepMind's AlphaGo, which defeated the Go world champion, and its follow-up program AlphaZero, has sparked a renewed interest of the general public in computational game playing. Moreover, game AI researchers build upon these results to construct stronger game AI implementations. While there is high enthusiasm for the rapid advances to the state-of-the-art in game AI, most researchers realize that they do not suffice to solve many of the challenges in game AI which have been recognized for decades. The Dagstuhl Seminar 19511 "Artificial and Computational Intelligence in Games: Revolutions in Computational Game AI" seminar was aimed at getting a clear view on the unsolved problems in game AI, determining which problems remain outside the reach of the state-of-the-art, and coming up with novel approaches to game AI construction to deal with these unsolved problems. This report documents the program and its outcomes.

Seminar December 15–20, 2019 – http://www.dagstuhl.de/19511
2012 ACM Subject Classification Computing methodologies → Artificial intelligence, Computing methodologies → Machine learning, Software and its engineering → Software organization and properties, Software and its engineering → Contextual software domains, Software and its engineering → Virtual worlds software, Software and its engineering → Interactive games
Keywords and phrases artificial intelligence, computational intelligence, game theory, games, optimization
Digital Object Identifier 10.4230/DagRep.9.12.67

1 Executive Summary

Pieter Spronck (Tilburg University, NL)
Jialin Liu (Southern Univ. of Science and Technology – Shenzen, CN)
Tom Schaul (Google DeepMind – London, GB)
Julian Togelius (New York University, US)

The past decade has seen a rapid advent of new technologies in computational game playing. For a long time, artificial intelligence (AI) for 2-player deterministic board games was mainly implemented using tree search, employing the minimax algorithm with alpha-beta pruning, enhanced with some improvements which were often aimed at particular games. This approach worked well for most traditional games, but some games proved to be notoriously hard to tackle in this way. The textbook example of games for which regular tree search is inadequate, is Go.

Ten years ago, the novel technique of Monte Carlo Tree Search (MCTS) became popular, as it was shown that using MCTS, the quality of AI for Go improved significantly, albeit not yet to the level of top-level human players. Many experts predicted that around 2030 Go AI would surpass human-level play. Much to the surprise of many, however, already in 2016 Google's AlphaGo defeated the human world champion in Go, using a combination of MCTS and deep convolutional networks to evaluate Go board positions and perform move selection. The networks were trained using millions of examples of human play, combined with self-play. A short while later, it was demonstrated with AlphaZero that self-play by itself suffices to train the networks to reach the necessary quality.

There is a long history of research into computational game AI for 2-player deterministic board games. However, since the turn of the century computational techniques have also been applied to games of a different nature, such as games for 3 or more players, games with imperfect information, and video games. Such games bring their own challenges, and often need very different approaches for creating game AI. Nevertheless, computational techniques may be applicable. Recent successes have been achieved in the playing of Poker (multiple players, imperfect information) and DotA (team-based video game). Deep learning techniques have been used to teach a game AI to play old Atari video games, and the highly complex game Doom, by only observing the screen.

These computational approaches to AI game playing have been highly successful, and have caused great enthusiasm in researchers and laymen alike. However, while they have opened up new possibilities for implementing strong game AI, they are definitely not the one-size-fits-all solution for all problems in computational game playing. The aim of the seminar was to build upon the foundations laid by the state-of-the-art in computational game playing, and (1) identify for which game AI problems the current state-of-the-art is inadequate or unsuitable, including the reasons why; (2) propose and investigate which improvements to the state-of-the-art may open up ways to apply it to a wider range of game AI problems; and (3) form ideas on which novel techniques may be employed to solve problems for which the current state-of-the-art is simply not suitable.

For the purpose of the seminar, a "game" is considered any simulated environment in which decisions can be taken in order to achieve a particular goal. This includes board games, card games, video games, simulations, and VR/AR applications. Decisions in a game are taken by "players." In multi-player games, the goal is usually to "win" from other players, by reaching a pre-defined victory condition before any other player manages to do so. "Game AI" is a computer-controlled player. Good game AI takes decisions which are highly effective in achieving the goal. Cooperative games are also of interest, where the aim is for the players to work together to share in a victory. We wish to point out that games are often a reflection of some aspects of the real world, and allow investigating those aspects in a risk-free environment – good solutions for problems found in games may therefore have immediate applications in the real world.

The following is a (non-exhaustive) list of challenges to game AI which are hard to deal with using the current state-of-the-art. These challenges formed the basis for the discussions and investigations of the seminar.

- **Determining the limitations of MCTS and deep learning for computational game playing**: The state-of-the-art in computational game playing encompasses Monte-Carlo Tree Search (MCTS) and deep convolutional networks to store game information.

The recent successes of this approach in Go have made MCTS and deep learning the "go-to" techniques for implementing game AI. However, these techniques have many inherent limitations. MCTS needs extraordinary amounts of computer power and is therefore very expensive to use. While it can be parallelized easily, just adding more computer power has diminishing pay-offs. Moreover, there are many games for which MCTS clearly is not a suitable approach, for instance, games with a large branching factor where it is hard to come up with heuristics which pinpoint the branches which are most likely to contain the strong moves. As for deep learning, now that the early enthusiasm has waned a little, the first criticisms of it, which explain its many limitations, are already being published. Gaining insight into the limitations of MCTS and deep learning for game AI implementation will allow us to distinguish those games for which these techniques may be employed for strong game playing from those games for which different approaches are needed.

- **Defining more appropriate game complexity measures**: Game complexity is a measure which is supposed to indicate how difficult it is to implement game AI. It is usually expressed as the number of possible game states in base log_{10}. Beyond a complexity of 100 (10^{100} game states), it is highly unlikely that a game will ever be "solved," i.e., will never be played perfectly. Researchers therefore aim for superhuman rather than perfect play. For a long time Go was considered the pinnacle of complexity in game playing, boasting a game complexity of 360. However, in the game AI domain, games have been researched with a much higher game complexity than Go. Typical examples of such games are:
  - *Arimaa*, a 2-player deterministic, perfect-information board game with a game complexity of 402.
  - *Stratego*, a 2-player deterministic, imperfect-information board game with a game complexity of 535.
  - *StarCraft*, a typical Real-Time-Strategy video game, with a varying game-tree complexity (depending on the parameters of the scenario) which is measured in the tens of thousands.

  The increased complexity of these games stems from multiple factors, such as an increased move complexity (e.g., in Arimaa players always make four moves in sequence), the introduction of imperfect information (e.g., in Stratego at the start of the game the players only know the location of their own pieces), or simply an explosion of pieces, moves, and non-deterministic influences (e.g., most video games). A common belief is that an increase in game complexity also entails an increase in difficulty of creating a game AI; however, previous investigations have shown that high game complexity does not necessarily equate high difficulty for achieving superhuman play. This indicates that "game complexity" might not be the most appropriate complexity measure for games. A theoretical investigation of game features may result in alternative ways to express game complexity, which may better relate to the difficulty of playing the game for an AI. Moreover, a better understanding of what makes a game difficult for an AI, might lead to new insights into how strong game AI can be built.
- **Learning game playing under adverse conditions**: In recent years, most research into game AI has moved towards "learning to play" rather than "implementing an algorithm." Game AI can learn from observing examples of human play (provided a large enough dataset is available) or from self-play. Such learning has lead to strong results in some cases, but in many cases fails under adverse conditions. Examples of such conditions are:

  - Imperfect information, i.e., the results of decisions of the AI depending partly on unknown data.
  - Continuous action spaces, i.e., the AI in principle being allowed to take an unlimited number of decisions in a small time period; thus, an AI not only has to decide what actions it wants to take, but also how many and with which intervals.
  - Deceptive rewards, i.e., situations in which positive results achieved by the AI in the short term, in practice drive it away from the ultimate goal of the game in the long term.

  To implement learning AI for wider classes of games, approaches must be devised to deal with such adverse conditions in systematic ways.
- **Implementing computational game AI for games with 3 or more players**: Most research into game AI is concerned with zero-sum 2-player games. The reason is obvious: in 2-player games, an objective "best move" always exists. With games that involve more than two players, which oppose each other, there often is no obvious "best move." For instance, if there are three players in the game, if two of those players band together, in general the third one will have a very hard time winning the game, even when taking into account that the other two are collaborating. The main problem is that when one player is obviously doing better than the other two, it is to the advantage of the other two to collaborate against the envisioned winner. Therefore, in games with three or more players, it may be advantageous not to play better than the other players, in order not to become the target of a collaborative assault of the opponents. A pessimistic perspective, where the AI assumes that the opponents will actually form a block, will in general lead to much worse play than a perspective wherein the AI tries to form collaborations itself. This means that the AI must incorporate in its reasoning the attitudes of the other players, for instance in the form of player models.

  The topic of AI for games of three or more players has been studied very little – a notable exception being the study of Poker, which is a relatively easy game in this respect considering the simplicity of the required player models and the very small state-space and game-tree complexities. Hundreds of thousands of games for three or more players exist, which by itself means that this challenge needs investigation. Moreover, when translating research findings to real-world challenges, the existence of more than two players is a given in most realistic situations.
- **Implementing AI for games with open-ended action spaces**: Certain classes of games have so-called "open-ended action spaces," i.e., the number of possible actions is basically unlimited. One example of such a type of game is found in interactive fiction: these are puzzle games which the player controls by typing sentences in plain English. While each game only understands a limited set of verbs and nouns, the player is generally unaware of this list. Designers of such games aim to allow the player to give any English command which is reasonable in the circumstances described by the game. Another example of such a type of game is a tabletop role-playing game, in which players are allowed to perform any action at all, and a game master determines (within boundaries of a complex ruleset) what the result of the action is. Creating an AI for either a game master or a player of such a game requires the AI to have at least a basic understanding of the game world to be successful. In practice, studies into game AI focus almost exclusively on games where the action spaces are closed, which makes regular learning algorithms applicable. For games with open-ended action spaces, as of yet no successful approaches have been found.

- **General Game Playing**: In the domain of General Game Playing (GGP), games are defined by a set of rules, specified in a General Game Description Language (GGDL). The goal of researchers in this domain is to create an artificial intelligence which is able to play such a game, based on only the rules. Yearly competitions are held where researchers pose their AIs against each other in games which are unknown to them at the start of the competition. The state-of-the-art in such competitions is using MCTS, enhanced according to some general assumptions on the types of games that need to be played. This approach is unsurprising, as MCTS does not require knowledge of the game in question in order to do reasonably well. In fact, this approach is so strong and so easy to implement that all competitors use it. The danger of the success of MCTS for GGP is that the research in this area gets stuck at a dead end – the same happened with the research into traditional game AI when for decades researchers only worked on small improvements to minimax and alpha-beta pruning, until MCTS came around to shake things up. It is highly unlikely that a blind and ostensibly "stupid" approach such as MCTS is the end-all of GGP AI implementations. It is therefore of particular interest to investigate novel approaches to GGP, which are not MCTS-based.
- **General Video Game Playing**: General Video Game Playing (GVGP) aims at designing an AI agent which is capable of successfully playing previously-unknown video games without human intervention. In the General Video Game AI (GVGAI) framework, video games are defined by a set of rules, sprites and levels, specified in a Video Game Description Language (VGDL). The VGDL was initially proposed and designed at the 2012 Dagstuhl Seminar on Artificial and Computational Intelligence in Games. The GVGAI framework has been expanded to five different competition tracks: (1) single-player planning, (2) two-player planning, (3) learning (in which no forward model is given), (4) level generation and (5) rule generation. In the planning tracks a forward model of every game is available; MCTS has been the state-of-the-art algorithm in these tracks. However, MCTS is not applicable to the learning track as no forward model is given and thus no simulation of game playing is possible. Deep reinforcement learning is a potential approach for the GVGAI learning track, but has not been investigated yet. Other methods might have potential too. Determining the applicability of different methods to the creation of GVGAI is a novel and topical challenge. Of particular interest in this respect is the creation of an AI for the domain of general Real-Time-Strategy (RTS) games.
- **Computation for human-like play**: Virtually all research into computational game AI focuses on building a game-playing AI which is as strong as possible. Strength can objectively be measured by pitting different AIs against each other. In video-game AI research, it has been recognized that playing strength is, in general, not a major goal – instead, much research in video game AI is aimed at making the AI play in an entertaining, interesting, or human-like manner. MCTS is notoriously unsuitable for selecting moves that are human-like, as it is simply based on finding the best outcome for the game as a whole. However, in situations where humans play against an AI, whether it is for entertainment or training, it is desirable that not only the best moves can be played by the AI, but also those moves which are interesting to explore or are on par with how a human might play. Almost no research has yet been done into computational human-like AI, which makes it a worthy challenge to take on.

The Dagstuhl seminar brought together computer scientists and industry experts with the common goals of gaining a deeper understanding of computational game AI, in particular to determine the limitations to the state of the art, to find new uses for the state-of-the-art, to explore new problems in the domain of computational game AI, and to investigate novel approaches to implementing computational game AI. Industry experts came not only from companies which specifically work in game AI research, but also from companies which use game AI in their products.

During the seminar we not only had discussions which investigate the topics theoretically, but also spent part of the seminar on trying to achieve practical results. We did the same in the 2015 and 2017 seminars, which was met with great enthusiasm and led to some strong follow-ups. As in the previous seminars, these practical sessions were partly to test out new ideas, and partly competition-based, where different approaches were used to implement AI for new problems, which were then compared to each other by running a competition.

What was new for this particular seminar, is that we held expert talks during some of the evenings. These started with one or two experts giving a longer talk (between half an hour and an hour-and-a-half) on one of their specialisms, followed by a longer Q&A session and a discussion. One evening was spent this way on using modern communication media to inform people about research (Tommy Thompson), one evening was spent on the details of DeepMind's AlphaStar (Tom Schaul), and one evening was spent on advanced search techniques in board games (Olivier Teytaud and Tristan Cazanave). These evenings were greatly appreciated by participants, and should remain part of this series of seminars.

Reports on the working groups are presented on the following pages. Many of these working groups have lead to collaborations, which will lead to papers to be published at conferences and in journals in the coming year. All in all, the general impression was that the participants and the organizers found the seminar a great success, and an inspiration for future research.

2 Table of Contents

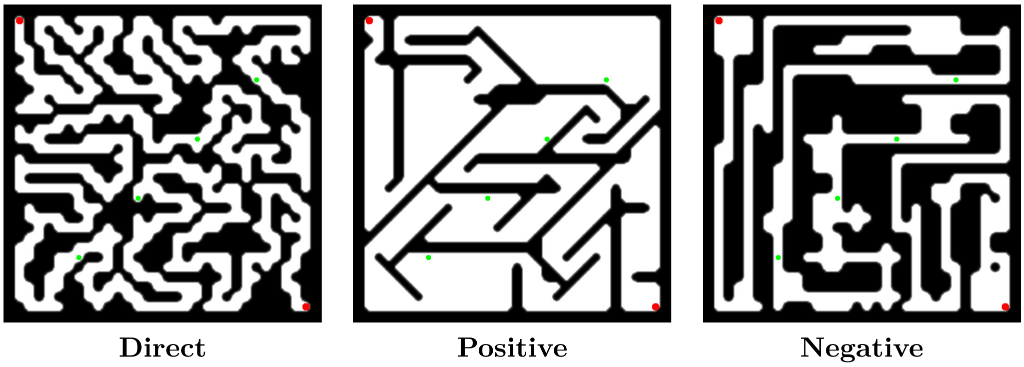


Figure 1 Three level maps evolved with the same fitness function but different representation. The fitness function maximizes the path length between the two red dots. The first image designates cells of the map as “full” or “empty” with a binary gene, the second specifies the placement of walls, while the third specifies where to dig out rooms and corridors.

3 Working groups

3.1 Representation in Evolutionary Computation for Games

Dan Ashlock (University of Guelph, CA), Tom Schaul (Google DeepMind – London, GB), Chiara F. Sironi (Maastricht University, NL), and Mark Winands (Maastricht University, NL)

Representation is a key feature of the design of evolutionary algorithms. This abstract looks at applications of the principle of the importance of representation to automatic content generation and games. Three topics were identified for further study: parameterized tree policy to permit a space of MCTS like algorithms to be studied, co-evolving games and agents to partition and understand game content generation spaces, and graph motifs as an enabling representation for the evolution of narratives to underly games.

Introduction

The topic of representation in evolutionary computation is very broad. The key point is that representation can have a large impact on the time required to solve evolutionary optimization problems and one the type of solutions located by an evolutionary algorithm system. Consider the problem of procedural generation of level maps. Figure 1 demonstrates the impact of representation on a simple procedural content generation problem [12].

The three example maps were all evolved to maximize the path length between a starting and an ending point, marked with red dots. The green dots were required to be in open space. The use of distinct representations caused a large impact on the appearance of the maps, demonstrating one of the impacts of choice of representation. Readers interested in additional reading on generating these sort of level maps might want to peruse [8, 1, 3, 4].

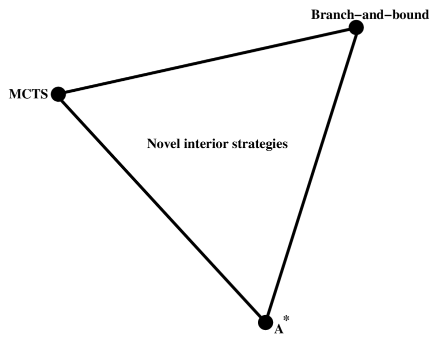


Figure 2 A diagrammatic representation of the parameterized tree space. The vertices of the triangle represent extreme settings of the parameters that recover particular tree-search algorithms that are the basis of the space.

Parameterized Tree Search

Monte-Carlo tree search [6] performs a directed partial search of a game space, functioning as an anytime algorithm that can refine its opinion about the correct next move until that move is required. The tree's policy is used to condition the balance between exploration and exploitation, favouring branches of the game tree that are promising or favouring branches of the game tree that are unexplored. Various *tree policies* are used for selecting which branch of the tree to follow and which new branch to try.

Many other tree search algorithms for games also exist [5]. The group identified the potential to write a parameterized tree policy that had MCTS, A$*$, branch-and-bound, and potentially other tree search algorithms, as the extreme cases of the space defined by the policy function. This notion is shown in diagrammatic form in Figure 2. Using such a parameterized tree policy opens the door to a novel type of research on tree policies. Problems that could be treated include:

- The relationship between category of game and appropriate tree search algorithm.
- The utility of the algorithms that arise in the interior of the space of possible tree policies.
- The identification of new tree search algorithms by considering how to induce new behaviours with the parameterized tree search policy.

The group also identified a potential modification of MCTS that might grant it improved performance. At present most tree policies are based on the quality of the branched of the tree and the degree to which they have been explored. Two added pieces of information might be useful: the depth within the game tree of the node of the tree currently under consideration and the number of roll-outs performed since the search was started after the last move made. The incorporation of these features creates the potential for tree polices that are sensitive to the stage of the game and the overall quality of the tree that they are extending.

Co-evolution of games and agents

A perennial, difficult topic is using computational intelligence to tune or create new games [11, 9]. The group identified a potential approach that would be interesting to try. When generating a new game or new pieces of procedural content there is a problem in that the fitness function used to drive evolution usually specifies much simpler tests that those implicit

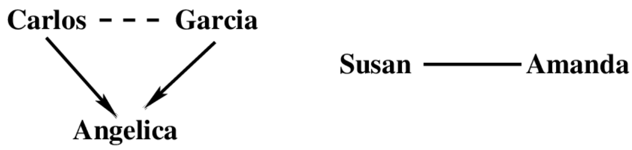


Figure 3 Two graph motifs. The first designates a love triangle in which Carlos and Garcia are competing for the affections of Angelina. The second indicates that Susan and Amanda are friends.

in the play of the new game or of the use of the automatically generated content. One approach to this is to attempt to enumerate the several factors affecting performance and utility and using multi-objective optimization [7]. This is a difficult approach if the numbers of factors being assessed are large.

An alternative is to use co-evolution of games or game elements with agents that use the evolved content. As long as the level of agent success with a game element can be used to generate numbers it is possible to create a system that imitates some of features of multi-criteria optimization. During evolution, an agent's fitness is their best score on any game or piece of game content. The fitness of the game or game content is the number of agents on which that game content has the best score. This system creates a situation in which agents compete to become more skilled and game content evolves to attract agents. This situation is similar to *partitioning regression* [2] in which models of data are co-evolved to create models that specialize in different parts of the data space. The partitioning of game elements and agents provides a collection of possibilities analogous to the Pareto-frontier provided my multi-objective optimization.

It is important to note that the evaluation function of the agents on the game or game elements must be chosen with some care to evoke the partitioning effect. An incorrect choice will lead to very similar game elements or one game element collecting all the agents.

Narrative generation via graph motifs

Another area of current research in games is the automatic generation of narratives underlying games [10]. The group identified the potential for search in spaces of social network graphs as a means of automatically establishing what actions a game engine should encourage to move a plot from an initial condition to a final condition specified by the game designer.

In this case, the social network graphs have vertices representing the players and non-player characters in the game as well as labelled edges that designate their relationship. Graph motifs are snippets of the graph that designate important or potentially important social situations. Figure 3 shows two graph motifs. In the first the arrows show interest, possibly romantic, or two young men in a young woman. The dotted line between the men shows that they know one another but are in competition. The second shows that two characters are friends. Characters and edges may participate in multiple graph motifs.

The game designer specifies and initial situation and a final situation. Search software is then use to connect the two social network graphs via a collection of *resolution operators* that transform motifs. With a rich enough collection of resolution operators, and the possibility of adding auxiliary characters to the scenario, almost any two graphs *can* be connected. The job of the search software is to find relatively short paths between the beginning and ending configurations as well as providing the game designer with a network consisting of multiple paths between the two configurations. This represents a rich space for manoeuvring the underlying narrative of the game toward the desired outcome.

Consider the Carlos, Garcia, Angelica triangle. Possible resolutions of this include the following.

- Angelica makes a choice, favouring one of the young men. The young men become {friends, strangers}.
- One of the young men kills the other to resolve the rivalry. Angelica {falls in love, is repulsed}
- An auxiliary character, Veronica is added to the social network graph and captures the affections of one of the young men.

Other possibilities are two characters who are friends of a third become friends, or possibly enemies. A friendship might sour. A romantic attraction may form. The key feature is that the space of motifs and the resolutions available for each motif must be rich enough to create a moderately rich network of social interaction graphs. This notion may be a bit difficult, but there is a network in which each node in the network is a social interaction graph, and the (directed) links in that network are resolution operators.

The possible relationships (edges), graph motifs, and resolution operators must be selected in a game-specific fashion, but the search engine can be generic once these quantities are specified. In a rich network space, evolutionary search is probably a good tool for traversing the network of graphs, which has a complex and not-too-structured fitness landscape. The group feels that development of this system could be quite valuable for narrative design in games.

References

1. D. Ashlock and L. Bickley. Rescalable, replayable maps generated with evolved cellular automata. In *Acta Physica Polonica (B), Proceedings Supplement*, volume 9(1), pages 13–22, 2016.
2. D. Ashlock and J. Brown. Evolutionary partitioning regression with function stacks. In *Proceedings of the IEEE 2016 Congress on Evolutionary Computation*, pages 1469–1476, Piscataway, NJ, 2016. IEEE Press.
3. D. Ashlock, C. Lee, and C. McGuinness. Search based procedural generation of maze like levels. *IEEE Transactions on Computational Intelligence and AI in Games*, 3(3):260–273, 2011.
4. D. Ashlock, C. Lee, and C. McGuinness. Simultaneous dual level creation for games. *Computational Intelligence Magazine*, 2(6):26–37, 2011.
5. D. Ashlock and C. McGuinness. Graph-based search for game design. *Game and Puzzle Design*, 2(2):68–75, 2016.
6. C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(1):1–43, 2012.
7. C.A. Coello Coello, González Brambila S., J. Figueroa Gamboa, and et al. Evolutionary multiobjective optimization: open research areas and some challenges lying ahead. *Complex Intelligent Systems*, 2019.
8. Lawrence Johnson, Georgios N. Yannakakis, and Julian Togelius. Cellular automata for real-time generation of infinite cave levels. In *PCGames '10: Proceedings of the 2010 Workshop on Procedural Content Generation in Games*, New York, NY, USA, 2010. ACM.
9. Jialin Liu, Julian Togelius, Diego Perez-Liebana, and Simon M. Lucas. Evolving game skill-depth using general video game ai agents, 2017.
10. Mark O Riedl and Robert Michael Young. Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*, 39:217–268, 2010.

11 Julain Togelius and Noor Shaker. *Procedural content generation in games.* Springer, New York, NY, 2016.

12 Julian Togelius, Georgios Yannakakis, Kenneth Stanley, and Cameron Browne. Search-based procedural content generation. In *Applications of Evolutionary Computation*, volume 6024 of *Lecture Notes in Computer Science*, pages 141–150. Springer Berlin / Heidelberg, 2010.

3.2 Game AI for a Better Society

Alan Blair (UNSW – Sydney, AU), Christoph Salge (New York University, US), and Olivier Teytaud (Facebook – Paris, FR)

The discussion in this group centered around two main themes: Game AI for Education, and Game AI for improved Mental Health.

There have been many efforts over recent years to develop specific games for vocational training, or to "gamify" components of the traditional academic syllabus.

Potential advantages of educational games include support for multi-view learning, scalability, equity of access and adaptability, promoting distinct capacities and complementary skills.

There is a diversity of opinion among educators as to whether students should be encouraged to play to their own strengths, leading to specialization; or to play to their own weaknesses, leading to good all-rounders. Studies have shown that when users are able to choose their own activity in educational games, those who already excel in abstract reasoning but not in social skills tend to choose strategic games like chess or checkers; and conversely, those with good social skills will often choose role-playing games. Counteracting this inherent self-selection bias may require careful incentives built into the system, or the intervention of a human instructor. On the other hand, the more important aim might be to empower the student and help them to better understand their own profile, and the effect of their choices on the coverage of their educational objectives.

One area in which Game AI could potentially have an impact is helping people to understand complex systems. Increasingly, a comprehension of phenomena arising from multiple interacting factors is needed in order to make even routine personal choices such as selecting a suitable mobile phone plan. Browsing effectively for accommodation, restaurants or leisure activities requires an awareness of the role of fake reviews and related phenomena. There are an increasing number of issues of public concern for which an informed citizenry with a general understanding of complex systems is necessary, in order to maintain a robust and healthy public discourse. These include electricity markets, university governance, fishing or water rights in a river system, economic and tax policies, anti-trust laws, geopolitical issues and trade agreements, as well as environmental issues and climate change.

Several existing games focus on issues such as Prisoner's Dilemma or Tragedy of the Commons. With these kinds of games there is a risk of becoming overly simplistic or didactic, or forcing a limited range of decisions. Ideally, these games should be constructivist and open-ended in nature, as well as being fun to play. One good example is the ECO game (play-eco.com).

Another role for Game AI might be in helping people to understand the possible impact of AI on society, and the economic, philosophical, practical and societal issues that may arise. One good example of this is the Fallout universe, which began in the 1980's dealing with issues such as robot terminators, and was expanded into the 2000's to deal with a broader range of issues including AI embodiment, slavery and fundamental rights. For example: To what extent should traditional human rights be extended to robots? Would it be acceptable to "torture" an AI?

Game AI raises many issues in relation to mental health. Modern tablet and smartphone games can be highly addictive and thus detrimental to the mental, physical and economic well-being of susceptible users. Perhaps, tougher legislative control is needed – particularly in relation to freemium games, loot boxes and games targeted to children.

On the other hand, Game AI could present an opportunity to improve mental health and wellbeing, by providing a companion agent, to replace or supplement a therapist, or to help people deal with depression, loneliness or alienation. One example of this is Ally from Spirit AI (spiritai.com) which is a conversational bot to help combat issues such as online bullying – including effective moderation, automatic intervention and behavior prediction.

There is a growing interest in the idea of using AI technology to make a positive impact on society, and we in the Game AI community should be looking for further opportunities to make Game AI part of this movement.

3.3 AI for Card Games with Hidden Information

Bruno Bouzy (Nukkai – Paris, FR), Yngvi Björnsson (Reykjavik University, IS), Michael Buro (University of Alberta – Edmonton, CA), Tristan Cazanave (University Paris-Dauphine, FR), Chiara F. Sironi (Maastricht University, NL), Olivier Teytaud (Facebook – Paris, FR), Hui Wang (Leiden University, NL), and Mark Winands (Maastricht University, NL)

This document sums up the discussion we had in Dagstuhl on Thursday 19 December about AI for card games with hidden information. We mostly discussed on the base of our ongoing experience on Skat and Bridge. Card games such as Bridge and Skat are Imperfect-Information Games (IIG).

In Bridge and Skat, a perfect solver with open cards is very useful. In Bridge, the perfect solver with open cards is named the Double Dummy Solver (DDS). A similar tool exists in Skat. Flat Monte Carlo (MC) using a Perfect Solver with open cards is the current state-of-the-art algorithm used by playing programs in Bridge and Skat. Another name for this algorithm is Perfect Information MC (PIMC). Flat MC may use simulators different from a perfect solver. Examples are the random simulator or any player which performs a game to the end, or evaluates a position.

In IIG, a node in the game tree can be an information set (IS). An IS is defined by the previous history visible so far by a given player. An IS depends on the viewpoint of a given player. An IS contains a set of states, each state containing the full information of the game at a given time. Because the player has partial information only, a given IS of this player contains all the states that this player cannot discriminate.

A problem identified with PIMC is strategy fusion. With an open card solver, the best strategy output by the solver assumes that each state has its best action, this action depending on the state it belongs to. Therefore, there is no reason that two states belonging to the same IS shares the same best action. This is the strategy fusion problem. However, an open card solver used in Flat MC remains the current best approach to Bridge and Skat.

Another issue in IIG is non-locality. You may consider two game trees: the IIG tree and the Complete Information Game tree (CIG tree). In Bridge or Skat, at the beginning of a game, the cards are dealt, then the play occurs on this specific card distribution. In the CIG tree, the root node is a chance node, and a search with open cards occurs in the corresponding sub-tree below this chance node. However, when searching with hidden cards in the IIG tree by performing actions over IS, the search meets IS with viewpoints from diffrrent players, and consequently, the search meets sub-trees of the CIG tree, different from the sub-trees in which the search started. This is the non-locality problem.

Recursive Flat MC (RFMC) was tried with success in Skat in 2013. RFMC has been tried recently on Bridge with success as well despite the computing time used.

In our discussion, we mentioned IS-MCTS, either with the Single Observer (SO) version, or with the Multiple Observer (MO) version. Various results were obtained with Flat MC and IS-MCTS on IIG. For games with more uncertainty, Flat MC was better than IS-MCTS. But on specific and small games, IS-MCTS could surpass Flat MC.

Alpha-mu is an algorithm which plays the role of the declarer in Bridge card play assuming the defense has perfect information. Alpha-mu solves the strategy fusion problem by playing the same action in all the states belonging to the same IS. Alpha-mu considers a set of worlds (card distributions) for which the set of outcomes are gathered in a vector. A set of vectors of outcomes (0 or 1) can be associated to a node in the tree. Specific operators on these vectors are used to back up information from nodes to nodes.

We also discussed about zero learning. Zero learning does not mean no learning, but learning without external knowledge, by self-play and starting with the rules of the games only.

We mentioned the importance of performing inferences in an IIG. We briefly discussed perspectives on Bridge and Skat, such as performing inferences with neural networks or solving the sub-game of winning the next trick in Bridge.

3.4 Human-AI Coordination

Katja Hofmann (Microsoft Research – Cambridge, GB), Duygu Cakmak (Creative Assembly – Horsham, GB), Peter I. Cowling (University of York, GB), Jakob Foerster (University of Toronto, CA), Setareh Maghsudi (TU Berlin, DE), Tómas Philip Rúnarsson (University of Iceland – Reykjavik, IS), Pieter Spronck (Tilburg University, NL), and Nathan Sturtevant (University of Alberta – Edmonton, CA)

Human-AI coordination is emerging as a new frontier for machine learning research. While historically a large part of the machine learning community has been focused on supervised learning tasks, new application areas require AI systems to collaborate and coordinate with humans in complex environments. Real-world applications include self-driving cars, virtual assistant technologies, game AIs, and household robotics.

Interestingly, much of the progress on multi-agent learning has focused on competitive settings, in particular two player zero-sum games. In these settings all Nash equilibria are interchangeable, so going from a self-play setting to playing optimally against a human opponent is trivial.

Coordination games are at the opposite end of the spectrum. In these games the optimal policy is crucially dependent on the strategy played by the other agents in the team. Yet, humans have remarkable skills at coming up with strategies that allow them to coordinate with an unknown team-mate on a novel task in a zero-shot setting. In contrast, there has been very little progress on this kind of problem in the machine learning and reinforcement learning community.

In this Dagstuhl seminar working group, we identified key directions for driving research towards more effective learning of human-AI coordination. First, since machine learning communities are benchmark and data driven, we believe that one reason for the lack of progress in this important domain is due to missing appropriate benchmarks. Second, we identified promising directions towards solving this challenge. Below, we briefly summarize these.

The team identified the following considerations for a benchmark for driving progress in human-AI coordination: (1) it should promote general solutions to avoid overfitting to individual specific game instances, (2) it should provide scope for collecting or providing data on human-human coordination, as a means of training and evaluation, and (3) it should provide a reliable measure of success that can reliably quantify progress towards human-like coordination.

Addressing these considerations, we propose the following benchmark setup. The key idea is to utilize a large number of novel *pure coordination* games, in which the optimal action crucially depends on the assumptions made about the gameplay of the (unknown) team-mate. One surprisingly entertaining example game is the *Guess 100* game, in which two players repeatedly choose numbers between 1 and 100 with the goal of converging to the same number as fast as possible. In this game humans typically play one of a number of heuristics that solve the game after a small number of guesses with an unknown team mate, displaying *zero-shot* coordination. Each of the proposed coordination games is accompanied by a large number of example game plays from humans playing online in an ad-hoc setting (ie. with a randomly assigned team mate) which can be used to develop new methods and to evaluate the performance of existing and novel learning algorithms. The goal of the benchmark would be to push for methods which produce human-like policies for the hold-out (test) games, without having access to the human game play data

Directions towards solving the proposed benchmark challenge include a wealth of research in game theory, machine learning, multi-agent systems, cognitive science, economics and many other areas. For example, one direction considers approaches that mimic the human ability to interpret the actions and intents of others, commonly called *theory of mind* (ToM). It involves interpreting the communicative intent when observing the actions of others, but also allows for finding strategies that can be interpreted successfully when observed by others.

Initial software to let AI play the *Guess 100* game, developed during the seminar, can be found here: https://github.com/katja-hofmann/dagstuhl-hac.

3.5 Evolutionary Computation and Games

Jialin Liu (Southern Univ. of Science and Technology – Shenzen, CN), Dan Ashlock (University of Guelph, CA), Günter Rudolph (TU Dortmund, DE), Chiara F. Sironi (Maastricht University, NL), Olivier Teytaud (Facebook – Paris, FR), and Mark Winands (Maastricht University, NL)

Evolutionary Computation (EC) techniques have been widely used in games. In this working group, we summarised the use of EC techniques in games mainly into the following categories:

- Planning in games: evolving action sequences for playing games [2].
- Agent training/adaptation: evolving/adapting (general) AI agents [5].
- Game design: level generation / puzzle design [3], game balancing [4], enhancing PCGML [6], design of NPCs.
- Evolutionary game theory [1].

In this working group, we brainstormed the pros and cons of EC for games, as well as the opportunities and problems. This abstract summarises the brainstorming discussions during the session.

An incomplete list of advantages and weaknesses of EC for games are listed as follows.

- Advantages:
  - **Easy to struct** Evolutionary Algorithms (EAs)[1] are easy to be structured and applied to different situations in games, e.g., for game design or game playing; single- or multi-player games; collaborative or adversarial games; with full or partial observation, etc.
  - **Derivative-free** EAs are derivative-free algorithms and meet well the needs of optimisation problems in game.
  - **Exploratory** EAs are powerful for their ability of exploring the search space.
  - **Robust to useless variables** EAs are robust to useless variables.
  - **Parallelism** EAs are population-based and can be easily paralleled.
  - **Diversity-convergence** EC takes into account both the diversity and convergence when evolving action sequences or game contents.
  - **Multi-objective** Multi-objective EAs can be used in the multi-objective optimisation problems in games.
  - **Partial observation** EAs can be applied to playing partially observable games by sampling.
- Weaknesses:
  - **Computationally costly** EC techniques can be computationally costly to find a reasonably good result (e.g., when using a single parent or having many restricted variables).
  - **Dense results** The results can be very dense and hard to understand.
  - **Representation matters** Designing good representation is hard. A bad one will significantly reduce the efficiency and effectiveness of EAs.
  - **Fitness** When there is no optimisation fitness, EC techniques are no more applicable.
  - **Hard to reuse data**

[1] In this abstract, we use the term "Evolutionary Algorithm (EA)" to refer to a whole family of algorithms, instead of a particular algorithm.

Among the above weaknesses, designing good representation and defining good fitness are particularly hard and important. For instance, when applying EC techniques to evolving game levels, how to design a suitable agent-based game fitness evaluation?

We agreed with the potential of using ECs for agent configuration, in particular for evolving selection strategies in MCTS.

References

1 Dan Ashlock, Mark D Smucker, E Ann Stanley, and Leigh Tesfatsion. Preferential partner selection in an evolutionary study of prisoner's dilemma. *BioSystems*, 37(1-2):99–125, 1996.
2 Raluca D Gaina, Adrien Couëtoux, Dennis JNJ Soemers, Mark HM Winands, Tom Vodopivec, Florian Kirchgeßner, Jialin Liu, Simon M Lucas, and Diego Perez-Liebana. The 2016 two-player gvgai competition. *IEEE Transactions on Games*, 10(2):209–220, 2017.
3 Ahmed Khalifa, Diego Perez-Liebana, Simon M Lucas, and Julian Togelius. General video game level generation. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016*, pages 253–259, 2016.
4 Jialin Liu, Julian Togelius, Diego Pérez-Liébana, and Simon M Lucas. Evolving game skill-depth using general video game ai agents. In *2017 IEEE Congress on Evolutionary Computation (CEC)*, pages 2299–2307. IEEE, 2017.
5 Chiara F Sironi, Jialin Liu, and Mark HM Winands. Self-adaptive monte-carlo tree search in general game playing. *IEEE Transactions on Games*, 2018.
6 Vanessa Volz, Jacob Schrum, Jialin Liu, Simon M Lucas, Adam Smith, and Sebastian Risi. Evolving mario levels in the latent space of a deep convolutional generative adversarial network. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 221–228, 2018.

3.6 Abstract Forward Models

Diego Perez Liebana (Queen Mary University of London, GB), Raluca D. Gaina (Queen Mary University of London, GB), and Tommy Thompson (AI and Games – London, GB)

Statistical Forward Planning (SFP) algorithms are those methods that use a Forward Model (FM) to plan an action or a sequence of actions from a given game state. Examples of these methods are Monte Carlo Tree Search (MCTS) and Rolling Horizon Evolution (RHEA). One of the main benefits of SFP approaches is that they are able to make new plans every time a new decision needs to be made, giving them flexibility and adaptability to face unexpected and previously unseen circumstances. MCTS and RHEA have been widely used in games research, from general (video) game playing [1, 2] to state-of-the-art approaches in Computer Go [4], Shogi, Chess and Atari [5]. Despite their popularity, however, SFP methods have rarely reached the games industry, with not many commercial titles using these methods. Notable exceptions are Creative Assembly's Total War series and AI Factory's card and board games.

One of the main reasons for the small uptake of SFP methods is their need for an FM. In particular, an FM needs to provide two pieces of functionality: a *copy* function, which creates exact copies of a game state, and a *next* function that provides a possible next game state given an original one and an action to apply to it. MCTS (or RHEA) normally select better moves if the number of iterations (or, respectively, generations) is higher, which is heavily influenced by the computational cost of the *copy* and *next* functions. Many commercial games (especially the ones that would benefit from the use of SFP methods) have complex mechanics and large game states, making the use of FMs unaffordable.

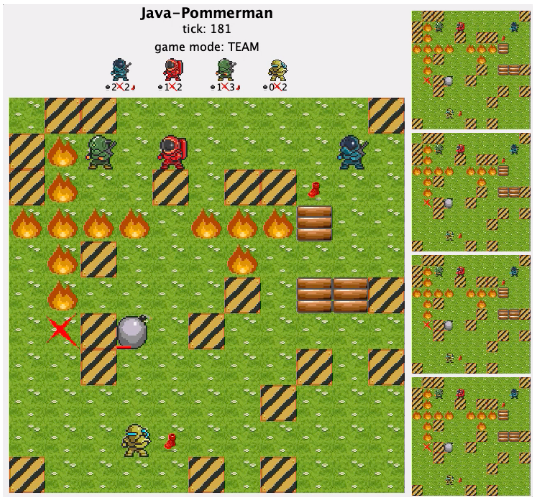


Figure 4 Java Pommerman framework.

This workgroup focused on approaching this issue by studying how *abstract* forward models can be created. Rather than learning forward models (as done in [5] for games of reduced complexity and size), the objective was to analyze how game states can be simplified into models that SFP methods can use without incurring in a significant loss of performance. In order to test this, we used a Java implementation of Pommerman, a 4-player simultaneous move game in which agents try to be the last one standing to win. Figure 4 shows a screenshot of this game. For more information about this game, the reader is referred to [3].

In order to test in the conditions of slow FM models, we artificially included delays in the different update rules of the game, such as computations for bomb explosions and movement, flame propagation and avatar collision checks. This created a version of the game that, because of the time required to carry out FM calls to the function *next*, was unusable by SFP methods. Then, we included the possibility for MCTS to decide i) which rules should actually be updated by the FM; and ii) a distance *threshold*, from the player's avatar, beyond which game state information is ignored by the FM's *next* calls. In effect, this created the possibility for MCTS to use an abstract (incomplete, inaccurate) FM that was faster to execute than the perfect one.

Our initial tests showed some interesting results. We ran MCTS players that use an abstract FM (AFM-MCTS) against the one that uses the complete and expensive FM (simply MCTS). They played in Pommerman's TEAM mode, 200 repetitions per pair. Two versions of AFM-MCTS where used: `Threshold-3`, which uses an FM that updates all the rules of the game except those occurring farther than 3 tiles away; and *Incomplete*, which uses an FM with infinite threshold where most game rules are not triggered at all.

For the `Threshold-3` case, the AFM-MCTS method was able to achieve a higher winning rate than the same MCTS algorithm which had access to a perfect FM. Simply by limiting the rules that can be updated to those taking place in close proximity to the agent, MCTS was able to run more iterations and therefore increase the quality of action selection, leading to a higher victory rate (62%, with 0.5% draws). When the FM abstraction is more severe (as in *Incomplete*, where some rules do not trigger in any case), the AFM-MCTS method victory rate falls, but only to 44% (1% draw rate). Considering this abstract FM is highly dysfunctional, it is remarkable that MCTS still achieves a decent winning rate using it.

In a way, this work shows how it is possible to *inject* Partial Observability to the model used for planning without heavily penalizing the performance of the agents. Which by itself opens interesting lines of future work. For instance, it would be interesting to investigate if it is possible to automatically detect which rules should be updated, and when. It could be possible to define priorities for these rules, which may depend on distances or ad-hoc heuristics. It could also be possible to statistically model (or learn) those expensive feature updates that are omitted from the FM, creating a hybrid between abstract and learnt FM.

In general, this work proposes two lines of research for the future. First, understanding how models can be simplified to a minimum expression without reducing the performance of the AI agents. Can these models be learnt automatically? What can we learn from a game design point of view, if some features can be ignored for AI decision making? Is it possible for these models to maintain a symbolic representation, hence improving the explainability of their decisions? Secondly, it is also worth investigating how SFP methods can themselves be improved to work with an abstract and inaccurate FM. This would not only aid in the use of abstract FMs for commercial games, but also in applications of automatically learning FMs with model-based reinforcement learning approaches.

References

1 Michael Genesereth, Nathaniel Love and Barney Pell. *General game playing: Overview of the AAAI competition*. AI magazine, 26:2, 62-73, 2005.
2 Diego Perez-Liebana, Jialin Liu, Ahmed Khalifa, Raluca D Gaina, Julian Togelius and Simon M. Lucas. *General Video Game AI: A Multitrack Framework for Evaluating Agents, Games, and Content Generation Algorithms*, IEEE Transactions on Games 11:3, 195-214, 2019.
3 Diego Perez-Liebana, Raluca D Gaina, Olve Drageset, Ercüment Ilhan, Martin Balla, and Simon M Lucas. *Analysis of Statistical Forward Planning Methods in Pommerman*, Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 15:1, 66-72, 2019.
4 David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche and others. *Mastering the Game of Go with Deep Neural Networks and Tree Search*. Nature, 529:7587, 484-504, 2016.
5 Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt and others. *Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model*. arXiv preprint arXiv:1911.08265, 2019

3.7 The Leonardo Project: Open-ended skill discovery

Sebastian Risi (IT University of Copenhagen, DK), Alan Blair (UNSW – Sydney, AU), Bruno Bouzy (Nukkai – Paris, FR), Nantas Nardelli (University of Oxford, GB), Tom Schaul (Google DeepMind – London, GB), Jacob Schrum (Southwestern University – Georgetown, US), and Vanessa Volz (modl.ai – Copenhagen, DK)

One of the open problems in artificial intelligence is open-endedness. The question is how can we create a system that, simililary to evolution in nature, can create an unlimited variety of different innovations.

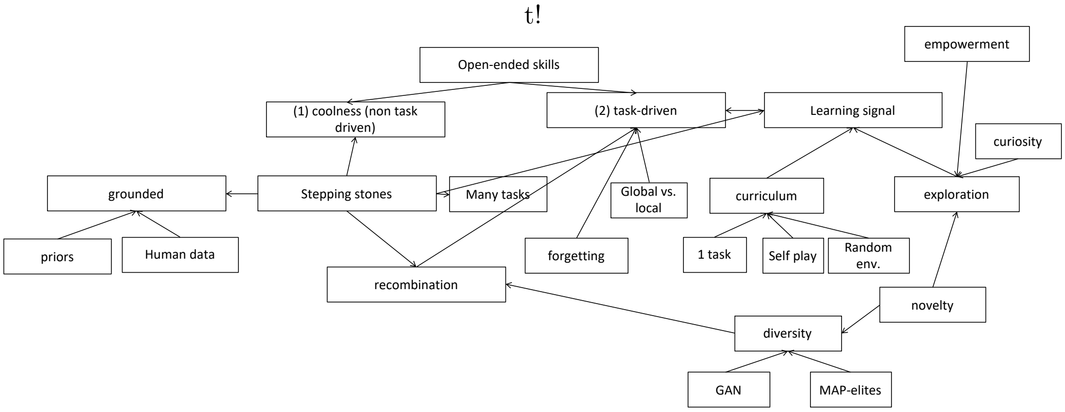


Figure 5 Overview of the different aspects of open-endedness.

While approaches modeled after evolution in nature (i.e. evolutionary algorithms (EAs)) have shown promise in optimization problems, no current EA does produce innovations in a truly open-ended fashion.

Overview of Open-ended Approaches

We sketched the landscape of open-ended approaches in Figure 5. First, approaches to OE can be divided into approaches that ultimately want to solve a certain task and approaches where the final task does not matter.

Additionally, the current approaches to discover diversity can be divided into the following approaches:

1. Co-evolving or complexifying environments
2. Self-play (curriculum but no diversity of skills)
3. Imitation Learning: Learning from human demonstrations (might not work all the time; need enough data)

Promising current approaches

One of the first examples of the idea of open-ended innovation is Minimal Criterion Coevolution (MCC) [1]. In MCC both the agent and the environment co-evolve to solve increasingly more difficult mazes. Recent work building on these ideas is POET [3], which deals with the more challenging OpenAI gym bipedal walker domain. POET is a good example of an approach in which solutions to a particular obstacle-course can function as stepping stones for solving another one. In fact, for the most difficult environments it was not possible to directly train a solution; the stepping stones found in other environments were necessary to solve the most ambitious course.

However, neither MCC nor POET are fully open-ended. Both approaches are restricted in the type of environments they can generate, and the goals the agents pursue. Some creative problem solving occurs, but the space of actions/behaviors in these example domains is fairly restricted. Therefore, it is worth considering what types of domains can actually support open-ended skill discovery, and what algorithmic ingredients would make it possible.

Necessary ingredients for Open-Endedness?

Several existing algorithmic tools may contribute to the discovery of open-ended behavior. Ones worth considering are listed below.

1. Reinforcement Learning
2. Quality Diversity
3. Co-evolution
4. Domain randomization
5. Populations
6. Curiosity-driven Exploration

The success of RL, and Deep RL in particular, in various domains makes it a technique that is impossible to ignore in the search for interesting behaviors. However, narrow pursuit of maximizing rewards in a particular domain seems to often converge to a narrow range of behaviors. Population-based approaches provide a way to store a variety of interesting behaviors across different individuals rather than inside of a particular agent's policy, but can also be susceptible to convergence. However, techniques such as Quality Diversity and Co-evolution encourage new behaviors to constantly emerge in evolving populations. Curiosity-driven Exploration can also explicitly seek new and different behaviors. Finally, domain randomization could also force an agent to support multiple behaviors, because a variety of unexpected domains means a variety of behaviors may be required to perform well. There may be interesting ways to combine these techniques in yet unseen ways to encourage open-ended behavior.

A new domain to study open-endedness

Here we propose a new domain to test the ability for open-ended innovation of different algorithms. In this domain, called the *Leonardo Project*, the agent has to try to learn to paint any painting with arbitrary given constrains. Given the success of other recent computational art approaches [2], it is important to emphasize that the new goal here is to have a single learning/training process that produces, in a single run, the ability to generate a wide variety of art in many different styles.

References

1 Brant, J. C., and Stanley, K. O. Benchmarking open-endedness in minimal criterion coevolution. In *Proceedings of the Genetic and Evolutionary Computation Conference* (2019), pp. 72–80.
2 Ganin, Y., Kulkarni, T., Babuschkin, I., Eslami, S., and Vinyals, O. Synthesizing programs for images using reinforced adversarial learning. *arXiv preprint arXiv:1804.01118* (2018).
3 Wang, R., Lehman, J., Clune, J., and Stanley, K. O. Paired open-ended trailblazer (poet): Endlessly generating increasingly complex and diverse learning environments and their solutions. *arXiv preprint arXiv:1901.0175* (2019).

3.8 Applications of Artificial Intelligence in Live Action Role-Playing Games (LARP)

Christoph Salge (New York University, US), Mike Preuß (Leiden University, NL), Spyridon Samothrakis (University of Essex – Colchester, GB), Emily Short (Spirit AI – Oxford, GB), and Pieter Spronck (Tilburg University, NL)

This abstract covers the results of the working group on how artificial intelligence techniques, particularly those commonly used in AI for Games, could be applied to Live Action Role-Playing (LARP) games and similar experiences. The results have been expanded on in a paper which has been submitted to a high-level conference.

Introduction

How could an artificial intelligence help a pretend paladin hunt an orc through a forest? The research field of Artificial Intelligence in Games, which should technically be able to answer this question, separates into two sub-fields. One dealing with how to make AIs that can play games to win, the other sub-field asking how AI can enhance the game experience. For both fields it is an interesting exercise to expand the scope of what a game is. Live-Action Roleplaying (LARP) is usually considered a game, or game-like experience, and it is gaining in popularity. In 2001 there were an estimated 100,000 active LARP players worldwide. In 2019 the German LARP calendar[2] lists 517 public events, with sometimes hundreds to (in rare cases) thousands of participants.

Using an AI to play such a game well seems to be a somewhat far-fetched endeavor – but as past AI predictions have shown, one should be careful in prophesying. It is imaginable that some day we will have a fully embodied AI that convincingly plays make-believe in a shared imagined world, navigating both the physical and the interpersonal challenges, while figuring out how to actually "win" in an open-ended, game-like interaction.

For now, we are more interested in AI applications to enhance the game experience in LARP. AI in games research in the past has also focused on game design, believable characters, world building, story telling, automatic game balancing and player modeling [10]. Naively, AI might seem like a poor fit for a game genre that is often associated with a deliberate lack of modern technology. But there are already early attempts to integrate modern technology into LARP [6, 7, 2].

We argue that there is a role for Artificial Intelligence in LARP – especially when focusing on those AI technologies that have already been successfully applied to other game genres. Furthermore, we think that the unique properties of LARP open up some interesting research directions for game AI. Moveover, AI can open some new possibilities for LARP gaming, e.g., super-LARPs that are hardly imaginable without AI. We should note that, going forward, we use the word AI rather loosely, and inclusively, to refer to a range of technologies from machine learning and neural networks to rule based systems and statistics. We also include those technologies resulting from such approaches, such as natural language processing.

[2] http://www.larpkalender.de/

Figure 6 Illustrative photo depicting a scene from a typical fantasy LARP. Players in costumes act out a scene in front of a backdrop of forest and period tents. Photo by anonymous, with permission.

Overview

The most common form of LARP sees participants come together, dressed as characters from a fictional world, and interact with each other in a live setting (Fig. 6). A LARP can be explained as an improvisational theater play in which you are playing a character, know about the other characters and the world, but do not have a script to follow [9]. There are many other traditions and corresponding descriptions of what LARP is. [3] provide a good overview. Central are usually a physical embodiment of at least some players as their character.

One interesting, and for our purposes relevant, distinction between table-top (role-playing) games and LARPs is the decomposition and realization of the different functions of the game master. In the first the game master (GM) usually interacts verbally with a group of players, sitting at the same table with them. The GM acts as a storyteller, quest provider, information provider, arbiter of rules, world simulator, and often also host. A lot of those functions operate on the fictional world that resides in the GMs mind, and players can interface with this world verbally. By stating their actions or speaking they affect this world, and the resulting effects are usually relayed to them with speech by the GM, all through one person. In LARP, particularly in larger ones, these different functions can be performed by different crew members, or in some cases other players, or the actual physical world. It is also typical that the different members in the organization team have different roles and responsibilities. While some crew members might be in charge of the games plot or overall event details, others might just be there to monster [8], i.e. play opponents and non-player characters, without having a understanding of the overall game or plot. This decomposition

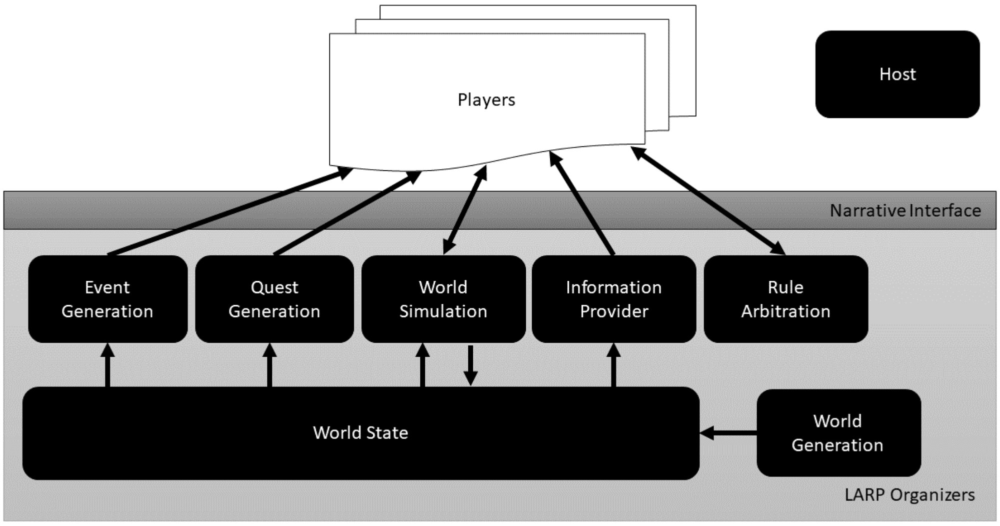


Figure 7 Decomposition of different functions / roles for LARP organizers. Arrows denote flow of information. In tabletop role play all functions in the grey box are usually performed by the game master, a single person, and all interactions are mediated by a single, narrative interface. The interface in LARP is between each function and the players, and can take different forms.

provides a challenge, but also an opportunity to solve different aspects of the AI game master problem separately. Fig. 7 provides a diagram of the functions we now describe in detail. We should note that this is largely oriented at big, entertainment focused, UK-style LARPS. Other LARP traditions may have similar roles, but their functions or limitations might be slightly different.

World State

The game world is at least partly fictitious. If a player were to kill an non-player character, then the character's death would be part of this fictitious world state. Facts like, for example, the identity of the chancellor of a fictitious in-world nation is also part of this world state. All the relevant information about the game world has to be (a) generated in the first place, (b) kept consistent with actions taken by the player and events in the world, and (c) used to inform events that happen. This data must generated at the start of the game, and be stored and synchronized, sometimes over multiple sessions.

World Simulations: When players act in the world, the effect of their actions has to be determined. Some actions of the player have to be perceived and acted upon by the organizers. This kind of world simulation, or effect on the world, is a typical bottleneck. The more players are participating in a given LARP, the more difficult it becomes to feed back all this information on the larger world state, and react to it appropriately.

Event Generation / Storytelling: Many LARPs feature a plot presented to the players in the form of events. The scale of LARP makes it hard to produce events tailored to specific players. As a result, many LARPs often have one person who keeps track of the current state of the world, decides on most major events, and integrates new information into the world model. That person is often overworked, struggling to keep track, or required to make design decisions. The limits of a single person's memory and speed of decision-making puts an upper bound on how much the players can influence the world and the planned order of events.

Quest Generation: In LARPs players often get quests. It is difficult to provide different quests for all people. A common quest for a group, or even all people is common. To compensate, players often come up with their own personal quests, similar to how real people might decide on life goals. Self-given, intrinsic goals tend to be of a different nature than extrinsic goals given by the game master. Quest generation ideally should be connected to the overall state of the world: quests should make sense, or be responsive to what is happening in the world, or their success should update the world state.

Information Provider: A common role for non-player characters (NPCs) is to provide information about the world to the player. Players are usually only aware of what they observed, so it is important to inform them about events or relevant facts. Information can be provided by documents or objects, but the most common approach is to have a NPC that brings up the relevant information "naturally" in conversation, or is specifically sent somewhere to inform players. This means that these NPCs need to be briefed.

Rules Arbitration: Most LARP events rely on an honor system in which player self-police their compliance with the game rules. For example, it is common that players have life points, and count themselves how many they lose in any given fight, determining themselves when they die. Nevertheless, there are usually a given number of referees that are there to arbitrate in case of conflict. Rules arbitration is particularly needed when hidden information is involved.

Hosting: In addition to managing the game's fiction, a LARP organizer is usually also responsible for hosting the event. This might include ensuring that there are adequate sleeping, food and hygiene arrangements, and that everyone is feeling safe and comfortable. This creates obligations to ensure a space that is free of dangers, harassment, etc. As with other games, it should be possible to withdraw from the game at any point without any negative real world consequences.

Summary: Decomposition: The described roles are not exhaustive, and they do overlap and interact to a certain degree. In smaller LARP events, several roles might be carried out by the same person. Nevertheless, the decomposition of roles is a distinguishing feature of LARP, and gives rise to some of the following challenges and opportunities:

Physicality

Most of a LARP is carried out in the physical world. This entails that real world phenomena interact with the fictional world and the world state.

Coherence

In an ideal world every meaningful player action updates the world state, and this update is communicated to all relevant parties. Such level of responsiveness is often not feasible for LARP. Because there is no objective external reality against which this information can be verified, wrong information propagates more within the LARP than it would in real life. Communication technology can address many of these issues. However, organizers often design their games to combat this decoherence – sometimes by providing fictional reasons why players cannot influence world state too much, and NPCs do not have more than a limited amount of information about the world.

Scalability

Scaling up a LARP often means adding more people to organizing roles, or splitting the roles among more people, and hence increases the challenge of keeping everyone in sync. It also means that more (coherent) content must be created to keep the game interesting for all players.

Immersion

There is an aim to keep a LARP immersive by removing everything that is counter to the world that is being presented, i.e., modern technology might be banned, even though such technology can be used to help the organization. Devices can be camouflaged as something else, and the often present concept of "magic" can be used to explain modern equipment or capabilities. It is also possible to have technology work in the background, and then design an appropriate interface that connects that technology to the game and the players.

Robustness

In many LARPs there is an understanding among players that they are not just consumers of an experience, but are actively helping to create the same experience for others. Player are usually willing to overlook minor problems and help to make LARPs work. This comes both in the form of being willing to adapt and interpret inconsistent clues, and in a willingness to improvise to fill the gaps. This gives LARPs a certain degree of robustness.

Possible AI Applications

In this section we want to present some sketched out examples of how existing, or conceivable AI approaches, particularly those from the AI in Games domain [10, 11], can be applied to LARP. The goal is to both make organizing LARPs easier, by overcoming the previously outlined challenges, and to enhance gameplay in a way not possible without AI.

Conversational Agents

There is currently tremendous interest from the AI community towards building conversational agents – commonly known as chatbots. The technology is not mature yet to the point where general conversations can be had. How could these chatbots be used in LARP?

One common interaction between players and referee is asking for rules clarification, such as "Can I dodge epic damage?" This is a role that could be filled by a straightforward question-answering system. As this is an "out of game" interaction, such a system could be deployed on a smart phone, or similar device, and there would be no need to simulate a character for that AI. This would already free up a large chunk of the time spend by the organization team.

Taking this a step further would be a conversational system that could provide "in game" information, such as "Who is the current ruler of this place?" In a most simple case, this could also just be a digital device with an question-answering system trained on a fixed text corpus, detailing the background of the world. But there are opportunities for improvement here. To further the immersion the conversational AI could be imbued with character traits that manifest in the way it speaks. Taking this even further would be to employ techniques that give an appearance of agency. Interactions with the AI could lead to changes in its mood – and the forms of interaction available might depend on past interaction.

There is also increasing interest in the concept of "virtual humans" – persistent non-player characters who might interact with a player via a combination of games, VR experiences, and social media. The technology developed to support such characters for marketing, theme park, and entertainment applications might also be suitable for deployment in LARPs.

Embodied AI Agents

There are also opportunities to having the previously mentioned AI agents physically embodied in the world. For example, a smart phone could be stored in a talking magical book. It could use its GPS sensor to determine its current location, and then trigger certain interactions when it is carried into a certain area. A conversational AI disguised as a familiar might turn into a game companion that rides around on the player's shoulder.

Drama Management: AI director

Narrative designers of conventional video games often use a system of storylets or quality-based narrative, in which story events are triggered whenever some pre-conditioned state is reached in the game world. They write individual moments that they want the player to experience at some point, and then allow the system to select the point when those moments are best presented during a particular player's playthrough. LARP creators have written about building LARPs with similar gameplay beats in mind. In the case of existing LARPs, it typically falls to human GMs to determine when the moment has arrived to deliver a story beat, and there is little room for last-minute customization. An AI system able to track key elements of world state, however, would be able to select when to activate particular storylets, and potentially use grounding techniques similar to those used in video games to fill in elements of the delivery, customizing the story moment to the exact parameters that allowed it to be fired off.

AI Content Generation

Before a LARP is first run, there is usually a phase where the organizers create a world and setting. There are two ways in which procedural content generation could help human designers with this. In a mixed initiative co-creative approach an AI system could produce fictional history or relationships, and a human designer could then select and refine. The system could also be used to generate inspiration of ideas.

On a more practical level an AI system could provide more complexity after the rough brush strokes have been filled in by a human designer. This might be useful to engage players who want to engage in a more scholarly play style. Once a system like this is set up, it would be easily scaleable – in theory one could provide a whole library worth of research that is both related to the world, and allows for relevant research to be played out.

Logic puzzles are a common element to LARPSs to serve as proxies for scholarly investigation. So a tablet representing a digital lock might require you to solve a sudoko or play a match three game to hack it. Existing AI techniques can already create a wide range of puzzles of a given type, allowing to add variability, or even design personalized puzzles in real time, related to the stats of the character impersonated by the player, or related to the world.

AI Story Hooks

Another opportunity for AI content generation is to produce story hooks or quests for players – providing for more micro-questing in larger LARPs. There are already existing approaches to automatic quest generation, usually looking at existing NPCs, their role in the world ontology, and their desires [4, 5, 1]. These tools could be easily modified to provide a range of quests to players, if a data based representation of the world and the players exists. These quests could then be offered in between events to players, as part of their event briefing package, or even prior to the event in digital form. This would allow the player to accept and reject certain type of quests – which could enable player modeling and more personalized quest generation. A player might, for example, not be interested in performing any illegal activity, as it clashes with their character concept.

AI-aided World Simulation

A big element of LARP, an a defining characteristic, is how the player interacts with the real and the game world. Many interactions of the players have immediate consequences provided by the environment and the other players. But there are usually also interactions with the fictional world where the player affects parts of the world that are not physically present. Here an referee usually has to interpret the players actions and determine how this affects the fictional, invisible world, and what consequences in they physical game world result from this. This is usually a massive bottleneck, resulting in design choices that aims to reduce the amount of this kind of interaction. AI could help by providing part of this complex world simulation.

Super LARP and Information Spread

Even more AI support than for single AI characters transferring information from Game Masters to players would be necessary if two or more LARPs would be connected in a way that they play within a common framework storyline. The more simple case would be succession in game time, so that one LARP takes up the open ends from a previous LARP. If, however, AI can support design and organization of a single LARP, there is no principal obstacle for several coordinated LARPs taking place at the same time.

Conclusion

In summary, we believe that LARP is a domain well suited for the application of AI and AI and Games techniques. The listed, existing approaches demonstrate this, and the speculative examples show a range of relatively straight-forward extensions of existing AI and Games techniques, so they would be suitable for LARP. Doing so could overcome several existing challenges for LARP organizers, such as scalabiltiy and content generation issues. It could also provide for new forms of play that would not be possible without AI. LARP also provides an interesting test-bed for AI applications, particularly those that want to explore the interface between humans and AI, or how AI can interact with the physical world. Here the robustness of LARP, caused by the willingness of the participants to correct errors on the fly, could provide valuable for researchers. In general, AI in LARP research offers several unexplored opportunities, both to enhance the experience of players, and to explore the limitations and challenges of AI.

References

1 Calvin Ashmore and Michael Nitsche. The Quest in a Generated World. *DiGRA Conference*, 2007.
2 Ella Dagan, Elena Márquez Segura, Ferran Altarriba Bertran, Miguel Flores, and Katherine Isbister. Designing "True Colors": A Social Wearable That Affords Vulnerability *CHI '19*, Association for Computing Machinery, New York, NY, US, 2019.

3 J Tuomas Harviainen, D Simkins, J Stenros, E MacCallum-Stewart, and D Hitchens. Live-action role-playing games. *Role-playing game studies: transmedia foundations*. Routledge, London, 2018.
4 Ben Kybartas and Rafael Bidarra. A survey on story generation techniques for authoring computational narratives. *IEEE Transactions on Computational Intelligence and AI in Games*, 9(3):239–253, 2016.
5 Lee Young-Seol and Sung-Bae Cho. Dynamic quest plot generation using Petri net planning. *Proceedings of the Workshop at SIGGRAPH Asia*, 47–52, 2012.
6 Elena Márquez Segura, Katherine Isbister, Jon Back, and Annika Waern. Design, appropriation, and use of technology in LARPs. *Proceedings of the 12th International Conference on the Foundations of Digital Games*, 1–4, 2017.
7 Elena Márquez Segura, James Fey, Ella Dagan, Samvid Niravbhai Jhaveri, Jared Pettitt, Miguel Flores, and Katherine Isbister. Designing Future Social Wearables with Live Action Role Play (LARP) Designers *CHI '18*, ACM, New York, NY, US, 2018.
8 Laura Mitchell. Volunteers as monstrous workers:'monsters' in UK live-action roleplay game organizations. *Culture and Organization*, 25(3):233–248, 2019.
9 Lizzie Stark. *Leaving Mundania: Inside the transformative world of live action role-playing games*. Chicago Review Press, 2012.
10 Georgios N. Yannakakis and Julian Togelius. A Panorama of Artificial and Computational Intelligence in Games. *IEEE* Trans. Comput. Intellig. and AI in Games, 7(4):317–335, 2015.
11 Georgios N Yannakakis and Julian Togelius. *Artificial intelligence and games* 2, Springer, 2018.

3.9 Procedural Content Generation for Minecraft

Christoph Salge (New York University, US), Alan Blair (UNSW – Sydney, AU), Alex J. Champandard (nucl.ai – Wien, AT), Michael Cook (Queen Mary University of London, GB), Raluca D. Gaina (Queen Mary University of London, GB), Diego Perez Liebana (Queen Mary University of London, GB), Jacob Schrum (Southwestern University – Georgetown, US), Emily Short (Spirit AI – Oxford, GB), and Vanessa Volz (modl.ai – Copenhagen, DK)

Main reference Christoph Salge, Michael Cerny Green, Rodrigo Canaan, Julian Togelius: Generative design in minecraft (GDMC): settlement generation competition. FDG 2018: 49:1-49:10

This abstract reports both on a talk and on a subsequent working group with the same title. The initial presentation was focused on the GDMC AI Settlement Generation Challenge in Minecraft [1]. This is a competition where participants write an algorithm that can generate an "interesting" settlement for a given, previously unknown Minecraft map. The general idea behind the competition is to put some focus of creativity and co-creativity as part of intelligence, and consequently as part of artificial intelligence. Currently, the main focus is on creating a settlement that fits into, and reacts to the environment given by an input map. The output should be similar to what a human could do. Further down the line there are plans to further extend the competition, to have AIs complete partially built settlements, and possibly do so from an embodied perspective. Currently, the AI interact with the map as a 3D representation. While the GDMC challenge hopes to stimulate research in this direction, it also aims to a.) solicit some ideas and approaches from the general public and b.) offer a joint framework where different technical approaches, such as wave form collapse or PCGML can be compared with each other.

The first part of the working group centered on discussing the details of the competition, its merits to science, and how to further extend and enhance the competition. Currently, all generated settlements are evaluated by humans, who are given a set of criteria – namely adaptivity, functionality, evocative narrative and aethetics. We discussed if those criteria could be automated – and how. Several of the working group attendees were actually expert judges for the competition, and shared their experiences. We also reviewed the results from the two previous years and looked at generated settlements from past competitions and discussed the used PCG techniques.[3]

We discussed different submission formats, and the newly added Chronicle challenge [2]. The goal there is to generate a text that is somehow about the generated settlement – for example, a historical narrative or a tourist guide. The main criterion for this is "fit", i.e. how well does a written text fit with, and only with, a certain settlement.

The second half of the working group was focused on coding up examples and trying out different approaches. Participants looked at various ideas, such as generating signs that reflect the features of the landscape, piping information from the settlement generation process into an existing travel diary generator, or developing a twitter bot that could walk around the settlement and take screenshots of interesting places and automatically tweet them out.

While there is no direct output planned for this working group, we should note that the GDMC competition is running again in 2020 (Deadline is the 30th of June), and several participants are considering to submit their own entries at this point.

References

1 Christoph Salge, Michael Cerny Green, Rodgrigo Canaan, and Julian Togelius. Generative design in minecraft (gdmc) settlement generation competition. In *Proceedings of the 13th International Conference on the Foundations of Digital Games*, pages 1–10, 2018.

2 Christoph Salge, Christian Guckelsberger, Michael Cerny Green, Rodrigo Canaan, and Julian Togelius. Generative design in minecraft: Chronicle challenge. page 5, 6 2019. 10th International Conference on Computational Creativity , ICCC 2019.

3.10 Black Swan AI

Spyridon Samothrakis (University of Essex – Colchester, GB), Nantas Nardelli (University of Oxford, GB), Günter Rudolph (TU Dortmund, DE), Tómas Philip Rúnarsson (University of Iceland – Reykjavik, IS), and Tom Schaul (Google DeepMind – London, GB)

Currently deployed AI systems has trouble acting on patterns what they were not trained with, though there are hints that humans can do that without much trouble. In this group we discussed how this might be possible and ways that machines could replicate human abilities.

[3] Material available online at http://gendesignmc.engineering.nyu.edu/

Introduction

The great strides we have seen in (Game) AI in the last 10 years or so seem to suffer from a serious lack of robustness; agents fail to learn concepts that will help them generalise in cases were the training distribution (which often comprises of carefully curated sets of data or example games) is different then the testing distribution (which could include a large subset of real world scenarios). One can make the case that there is probably enough information in the data to act reasonably in an out-of-distribution fashion, but the current crop of AI systems seem to fail at the task. Given that distributions of real-world events might be somewhat "flat", this poses serious problems, as there is no way that an agent could be provided with enough data that contains all the possible configurations of the known universe.

"Black swan" AI

We broadly and loosely define black swan AI[4] as systems that can act reasonably on a world configuration that they have not directly perceived during their training phase. For example a self-driving car that has never seen a pedestrian crossing the highway should be able to maintain internal coherence and deduce the right action (e.g. push the break pedal).

Training robust AIs

A serious issue posed in designing "black-swan" robust systems is the difficulty of designing correct testing procedure. Once a test scenario has been recognised, it is somewhat easy for a system designer to include the relevant inductive biases in their algorithm and effectively solve the specific problem instance, without addressing the bigger issue. A potential solution to this methodological problem would involve two researchers, a test generator and a test solution provider. The generator would create a novel problem on a weekly/monthly/annual basis and provide a subset of the data, while keeping an out-of-distribution test set hidden. A solution provider would then attempt to solve the problem and the two would compare the results for the test set only once, as to avoid any information spillage.

How do humans do it?

We postulate that humans can act on completely fictive scenarios, both in terms of state and action space. For example it is very easy for a human to imagine a battle between her and a TV-come-to-life, though the scenario is extremely unlikely (a television set cannot really fight). It also known that humans can base their decisions and societal organisation on completely fictive thoughts and abstract concepts (e.g. what is a corporation?). In that sense, no world permutation is unimaginable (and hence no black swans in the way we defined it above) – utterly outrageous counterfactuals can be acted upon.

[4] "Black Swan" as a term was popularised by Taleb [1]

How could machines possibly do it?

Though the training/test set problem identified above is a major issue, ideas that are closely aligned to procedural content generation[2] could prove helpful; one can treat certain environments as one of many possible instances of a generic concept (for example, the game of Mario as a specific realisation within a family of Mario-like environments, where Mario can take radically different actions (e.g. fly), while the shapes of obstacles change from square to random). Another possible way of thinking, especially in the context of vision, is learning computer vision by first learning computer graphics[3]. One could learn a generative model of graphics and go back to learning models of acting on generated instances, which can now be made to comprise of the totality of vision.

References

1 Nassim Nicholas Taleb. *The black swan: The impact of the highly improbable*, volume 2. Random house, 2007.
2 Noor Shaker, Julian Togelius, and Mark J Nelson. *Procedural content generation in games*. Springer, 2016.
3 Geoffrey Hinton. Taking inverse graphics seriously, 2013.

3.11 Learning to Learn

Spyridon Samothrakis (University of Essex – Colchester, GB), Bruno Bouzy (Nukkai – Paris, FR), Michael Buro (University of Alberta – Edmonton, CA), Setareh Maghsudi (TU Berlin, DE), Tómas Philip Rúnarsson (University of Iceland – Reykjavik, IS), and Tom Schaul (Google DeepMind – London, GB)

In this workshop we discussed the design of machine learning systems that learn their learning mechanisms (instead of just learning from data).

One can easily imagine a taxonomy of AI systems scafflolded alongside a path of recursive learnability. Systems of hardcoded rules would form the beginning of a ladder, while systems that learn would from data would be an immediate step up. An obvious and natural extension to a further step would be to have systems that learn how to learn.

This form of meta-reasoning has extensively studied in the literature. We broke down the problem into a set of *interfaces* and *objectives*. Interfaces highlight the method that a learning to learn algorithm would be implemented in, while objectives define goals that we would like to achieve by employing a learning to learn mechanism.

Overall the goal would be to search the algorithmic space defined by a learning process and adapt the learning mechanism. As such, one can easily envision a recurrent neural network being used as the learning mechanism for neural networks, replacing generic variations of SGD with problem specific learning mechanisms. Along the same line of thought, one can easily envision novel hebbian rules [1] being learned, again as mechanisms to achieve a certain objective.

The difficulty of designing good algorithms in this niche seems to lie in the fact that searching through a wide algorithmic space is hard (and possibly prohibitively computationally expensive). If these algorithms are to more broadly accepted as part of the standard AI toolbox, careful design of what is allowed to learn would need to take place.

Table 1 Examples of interfaces and objectives. Interfaces define the mechanism, while objectives define the goal of a learning to learn mechanism.

Interfaces	Objectives
Learning network connectivity	Sample efficiency
Learning to explore	Avoiding catastrophic forgetting
Learn initial weights	Robustness
Learning hyperparameters (e.g. γ, λ)	Reward maximisation
Learning the learning rate	Exploitation Maximisation
Learning the update rule	
Learning the opponent adaptation	

References

1 Joseba Urzelai and Dario Floreano. Evolution of adaptive synapses: Robots with fast adaptive behavior in new environments. *Evolutionary computation*, 9(4):495–524, 2001.

3.12 Self-Adaptive Agents for General Game Playing

Chiara F. Sironi (Maastricht University, NL), Dan Ashlock (University of Guelph, CA), Cameron Browne (Maastricht University, NL), Tom Schaul (Google DeepMind – London, GB), Hui Wang (Leiden University, NL), and Mark Winands (Maastricht University, NL)

When implementing General (Video) Game Playing agents that play a wide variety of (video) games, it is always hard to find a game-playing strategy or a setup for the agent that is optimal for all games in all circumstances. Therefore, the aim should be to design a game-playing approach that adapts various characteristics of the agent to each game being played. When designing self adaptive agents, multiple aspects should be taken into consideration. It is important to define the objectives to be achieved by adapting the agent. Moreover, we should decide which characteristics of the agent we want to adapt and whether the adaptation should happen on-line or off-line. Finally, we should consider that the structure of game trees might vary substantially from game to game. This increases the complexity of adapting agents to each game. These aspects are discussed next.

Objectives

First of all, multiple objectives can be identified towards which agents could be adapted. For example, the agent could be adapted to each game to increase its playing *strength*, its *efficiency* or its *robustness*. *Fun* is another possible objective. For instance, when playing against humans, agents could be adapted to the level of the human player to keep the game entertaining. Agents could also be adapted in order to *avoid delusion*. A simple example is when an agent believes to be winning the game because it is actually not searching the parts of the game tree where losing states are present. In this case, search could be adapted to each game to explore interesting parts of the game tree. Finally, another possible objective is *transparency*. A desirable feature of an adaptive agent would be the ability to explain why it is making certain decisions and why it is searching certain parts of the tree for each game it plays.

Approach

There are also many ways in which the agents themselves can be adapted, for example the *algorithm* they use to play each game. The agent could not only change the algorithm used to play, but it could also adapt the *parameters* that control the game-playing algorithm and also the *heuristics* that such algorithm is using. As a concrete example of adaptation of the search algorithm, Genetic Programming could be used to explore the space of action-selection functions, looking for the one that is most suitable for each game at hand. The performance of an agent on different games or at different stages of the game might also depend on how further ahead the agent is looking. Therefore, the search *horizon* could also be adapted. Another alternative consists in adapting the representation of the *actions* or of the *game model* or of the *rules*. For example, depending on certain characteristics of the game a representation might be more efficient than another one. Finally, for most of the games the search space is too large to be visited exhaustively. Therefore, agents could be designed to adapt the *sampling strategy* that chooses which parts of the tree to visit and how many times.

Timing

Another aspect to take into account is when the adaptation takes place. A wide spectrum of possibilities is available, that goes form fully on-line adaptation (i.e. while a single run of the game is being played) to fully off-line adaptation. An example of the first would be adapting the agent after each *trajectory* that is explored during the search. A less fine grained alternative would be adapting the agent after each *move* that is performed in the game. Moving towards the off-line end of the spectrum we can find adaptation to each *episode* (or game run), to each game *level*, to each *game* or to each game *genre*.

Complexity

One final aspect that should be considered is that in General (Video) Game Playing the domain structure is partly unknown. Moreover, game trees have a huge variation in branching factor and depth. Therefore, which technique works on which game might depend on the structure of the game tree. This means that there is the need to understand the game search space in order to be able to adapt the search and explore such space efficiently. An interesting direction for future research would be looking into how search spaces can be visualized. In addition, it would be interesting to visualize and analyze search trees. Investigating metrics that can be applied to search trees might enable us to discover desirable or undesirable properties in a game. These properties might help the process of adapting the agent.

3.13 NPC Understanding

Pieter Spronck (Tilburg University, NL), Duygu Cakmak (Creative Assembly – Horsham, GB), Jialin Liu (Southern Univ. of Science and Technology – Shenzen, CN), and Mike Preuß (Leiden University, NL)

Non-Player Characters (NPCs) in games exist in a computer-controlled game world, where they exhibit behaviors which they express through conversations and actions. Game developers use a high variety of techniques to create Artificial Intelligence (AI) which controls these NPCs. Common approaches include behavior trees, state machines, and straightforward scripts. The reason to employ such basic techniques is that computational power for running AI in a game is limited. Moreover, these techniques lead to AI which is easy to create, debug, and maintain.

A disadvantage of the classic approaches to implementing NPC behavior is that the NPCs only react in specific ways to events which happen. For instance, if a human player takes an item in the game which, according to the game, belongs to the NPC, the NPC might get angry and attack the player. This is because implicitly the game AI assigns a motivation to the human player for taking the item, labeling the human player as a "thief." However, the motivation of the human player might be very different. No games exist, however, in which the human player can express their motivation, or in which game AI derives motivations in a more intelligent way. For commercial video games, that might actually suffice, but for many serious games, it is necessary that plausible motivations for human player behavior are derived by the game AI.

To assign plausible motivations to players, NPCs need to have some rudimentary kind of "understanding" of the player's behavior in the game world. For example, consider the following situation: an NPC is sitting at a table and eating food from a plate, when the human player enters, grabs some food from the plate, and eats it. Depending on an understanding of the role of the player in the game world, different motivations can be assigned to the player. If the player is in a romantic relationship with the NPC, this might be considered reasonably normal. If the player is a good friend of the NPC, this might be considered a joke. If the player is a stranger to the NPC, this might be considered an insult. Proper responses might be a friendly smile, a confused look, or an angry scowl, respectively.

Ultimately, with a system of understanding, game AI may be able to make suitable predictions on what is supposed to happen after a human player interacts with the game world, and make suitable backwards predictions on how a certain situation has come to pass. For example, an NPC who finds the player, who is a stranger in the village, on their front porch, then discovers that their front door is unlocked and a valuable sword from their home is stolen, might not actually know that the player stole the sword, but backwards reasoning should provide him with at least enough of a suspicion to ask a local guard to frisk the player.

To supply an NPC with a rudimentary understanding of a game world, the following components are needed:

- An entity-relationship model which describes the objects in the NPCs world, and the relationships between the objects (e.g., buildings and their owners). This ER-model can be shared by all NPCs.
- A network of social structures (i.e., what social relationships exist between NPCs).
- A description of social norms, which assign a rating of 'normality' to each action in combination with the objects and social structures involved; if social norms are culture-dependent, then NPCs may have different sets of social norms.

- Optionally, an overview of the history of the game world.
- For each NPC, an internal state, including personal attributes and goals in the world.

Using these components, and a set of potential NPC actions, NPCs can generate a reasonable action list for each moment in time. From the reasonable action list they can choose an action to perform based on their state and attributes. Considering that without interaction of a human player, the 'life' of a group of NPCs will be repetitive, unchanging from day to day, using the set of components listed above, an expected world state can be generated and stored for each moment in a day. The expected world state is simply the state of the world which exists if only NPCs interact with the world.

The expected world state can be compared with the actual world state, and any discrepancies (which are probably caused by interactions of the human player) need to be interpreted. This interpretation is influenced by the place of the human player in the social structure, and an NPC's selection from the reasonable action list can be influenced by it. In a realistic implementation, if an NPC cannot derive a good motivation for a human player's behavior, it can get the option to simply ask for it.

The structure described above can also be used to generate new daily behaviors for NPCs when a player interferes dramatically with the game world. For example, suppose that in a virtual village many NPCs get their daily nutrition from eating a baker's bread, and the player kills the baker, then the NPCs can no longer achieve their goal of "having their hunger quenched" by buying and eating bread. Consequently, they will select different actions from their reasonable action list. If this system is implemented well, it could lead to a different stable state for the village, in which a new baker is appointed or villagers get their food from different sources.

Naturally, implementing a system as described above is a challenging undertaking, but in principle, since game worlds are so much simpler than the real world, a small-scale could be run. Such an experiment could be a lead-in to, in the far future, create AI which has an understanding of the real world, rather than a game world.

3.14 AI for Accessibility in Games

Tommy Thompson (AI and Games – London, GB), Bruno Bouzy (Nukkai – Paris, FR), Alex J. Champandard (nucl.ai – Wien, AT), Michael Cook (Queen Mary University of London, GB), Diego Perez Liebana (Queen Mary University of London, GB), and Emily Short (Spirit AI – Oxford, GB)

Despite there being approximately 164 million people in the United States playing video games, there has historically been a lack of support for players with accessibility issues. In this workgroup we discuss the open challenges yet to be explored and opportunities for artificial intelligence to be employed as means to enhance player experience.

Introduction

Video games have become increasingly more pervasive in modern culture, with an estimated 164 million people in the United States playing video games, accounting for 65% of the country [1]. However, while this figure is considerable, there has historically been a lack of support for people who have issues accessing games due to physical or mental health issues or disabilities. Given the nature of the medium and the input devices often required to interact with them, video games can present a variety of challenges for players due to pre-existing health conditions.

It is estimated that of the aforementioned 164 million people in the United States, 33 million of them are disabled [2]. This results in a need for different solutions to address the likes of reading text prompts in user interfaces, to interacting with 'dual stick' gamepad controllers on popular games consoles. At the time of writing, the range of accessibility options in games is slowly increasing. With new tools and peripherals such as the Xbox Adaptive Controller providing alternative input methods for common interfaces, to the introduction of hard requirements for video games to provide basic accessibility options in areas of communication [3]. Ranging from text-to-speech systems enabling for users to have menus and other text-heavy regions read out to them as seen in games such as *Tom Clancy's The Division 2* [5], to the ability to customise font size, colour and background for spoken dialogue and narration exhibited in *Gears 5* [6].

Despite this, many of the accessibility options available still fail to address many of wide-ranging and often bespoke issues that players face. New regulations on accessibility options introduced by the Federal Communications Commission in the United States have taken longer to appear in video games than they have done in other entertainment mediums such as film, and television. With the original regulation introduced as part of the '21st Century Communications and Video Accessibility Act' ("CVAA") in 2010, With a waiver applied to the video games industry until the end of 2018. While this waiver no longer applies to the industry as a whole, there are still dispensations afforded to smaller independent games companies due to the expected costs when compared to the budget of the project. Furthermore, the FCC regulation on deals exclusively with a subset of accessibility concerns in relation to communication. Hence many other concerns are still largely ignored.

Hence in this workgroup, we aimed to have a discussion about the state of artificial intelligence applications to accessibility in games. The areas within which AI can be applied and envisioned some potential applications based upon existing technologies and more novel applications that are within our capabilities as a research community.

Accessibility: Wants and Needs

The notion of 'accessibility' is rather broad and could be considered to encapsulate a variety of different users, including:

- Players who would enjoy additional features to enhance their experience.
- Players for whom accessibility features help remove issues that have a small impact upon gameplay.
- Players for whom accessibility features enable them to enjoy playing a given game. Without these features, it is highly probable they would not be able to play the game.

What is clear is that there is a difference of *wants* versus *needs* when identifying and designing potential accessibility tools. In the majority of cases, a tool that satisfies a need will cater to all audiences. A good example is use of subtitles for spoken dialogue. Given players with hearing loss need these solutions to understand what it happening, while others may prefer them to enhance their experience. This behaviour has been exhibited in recent video game releases, with publisher Ubisoft acknowledging 60% of players turned on subtitles in games released in 2017. However, when they made subtitles enabled by default in later releases, only around 5% of players turn them off [4].

Opportunities for AI

At present there are not many artificial intelligence applications, if any, that focus specifically on providing support to video game players with accessibility needs. However, when we consider specific problems areas, there are existing applications and research areas that could turn towards game-specific challenges. For example, text-to-speech (screen reader) technologies could be adapted to read the game world in a concise and useful manner for a player to make informed decisions. This brings its own challenges of understanding what information is pertinent to the player at that time and would make for an interesting body of research: given the needs and priorities of the reader would shift for each player and game that they play. Meanwhile input interpolation and prediction used in motion control could be adapted to better understand the intended movement from a users with limited motor functions.

Conversely, there are established problems emerging within the video games industry that could benefit from further research and development. Colour blindness filters for user interfaces are recognised as a desired feature within combat games, but there is little consensus on how to solve the problem. Filters are provided for known colour blindness types (deutronopia, tritonopia), but there is a lack of consistency from one game to another. Working with players through an interactive learning process to train the desired colour blind filter could be a useful feature – applied at the level of a PC or console operating system such that it can be applied and later modified or improved on a per-game basis.

Future Steps

While the authors are confident of the capabilities of artificial intelligence and the community's capacity to address these concerns, what we lack is a strong understanding of the problem. It would be presumptuous to initiate a concerted research effort into these problems without understanding what are the larger concerns of disabled players and those who support them. As such, further research in this area should commence by initiating discussions with the one or more charities and other registered bodies who provide support, resources and advice to disabled players. Two notable examples being the 'AbleGamers'[5] charity within the United States, as well as 'Special Effect'[6] in the United Kingdom. With a more thorough understanding of some of the larger issues faced by disabled players a more concise and informed position can be made on how to proceed.

References

1 Entertainment Software Association. *2019 Essential Facts About the Computer and Video Game Industry*, August 2019.
2 AbleGamers *The AbleGamers Foundation – Helping Disabled Gamers Everyday!*, YouTube (AbleGamers), 2012.
3 Frankfurt Kurnit Klein & Selz PC. *Video Games With Advanced Communications Services Must Now Be Accessible to Players With Disabilities*, January 2019.
4 GamesIndustry.biz. *Ubisoft Sees High Acceptance for Opt-Out Subtitles*. June 2019.
5 Massive Entertainment. *Tom Clancy's The Division 2*. March 2019
6 The Coalition. *Gears 5*. September 2019

[5] https://ablegamers.org
[6] https://www.specialeffect.org.uk

Figure 8 A game being played by us during the workshop.

3.15 Consistor

Julian Togelius (New York University, US), Jialin Liu (Southern Univ. of Science and Technology – Shenzen, CN), Mike Preuß (Leiden University, NL), Sebastian Risi (IT University of Copenhagen, DK), Jacob Schrum (Southwestern University – Georgetown, US), Tommy Thompson (AI and Games – London, GB), and Georgios N. Yannakakis (University of Malta – Msida, MT)

Consistor is a game where you must always take the same decision when faced with the same situation. In other words, you are only free to choose what action to take the first time you are faced with a new situation. Our workshop designed single- and two-player versions of the game and AI-based methods for refining the game. We foresee the use of search algorithms and evolutionary computation to help playtest and analyze the game. More (or less) interestingly, we also made connections between the central ideas of the game and certain ideas from ethics, psychology, and media studies.

Single-player Consistor

Firstly, we designed a single-player version, in which a player starts from a vertex of the game board and aims at reaching the vertex at the diagonal line. When the game starts, no rule has been defined, the player designs the game rules by making a decision at his/her first visit to each colour of tiles. And then, when a colour is visited again, the player should follow the defined rules. Figures 8 and 9 provides some illustrative examples. We have tested the game by generating boards randomly and moving the starting and target location around. In general, this game is easy to solve when the board dimension is low.

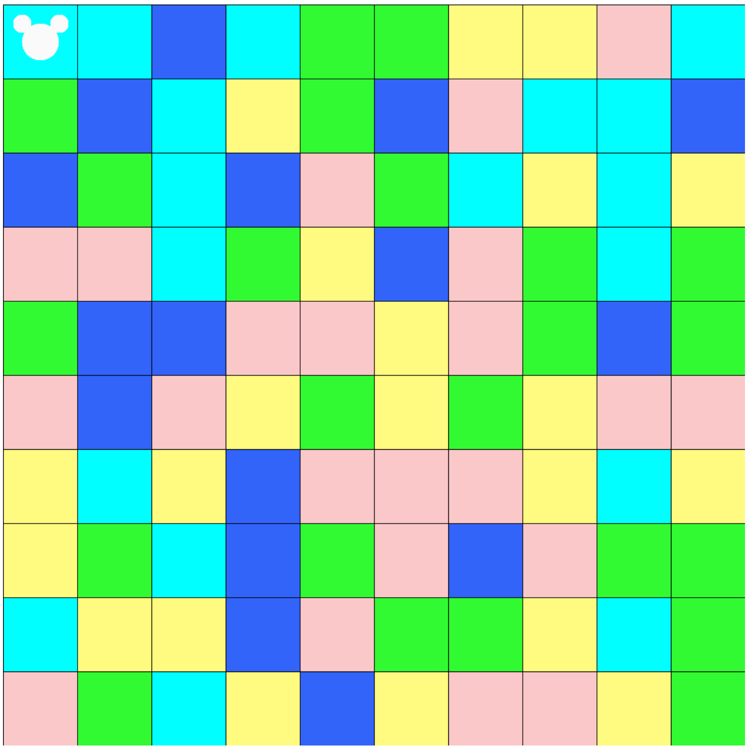


Figure 9 A randomly generated playable 10 × 10 board for single-player Consistor.

Two-player Consistor

It is straightforward to extend the Consistor game to a two-player version. We considered a simple cooperative version and a more complex version with a payoff matrix in order to imitate the ethics choices and and psychology conflicts in real life.

Cooperative Consistor

In the cooperative version, two plays start from different locations and take actions in turn, thus, they define the game rules together. The result of a game can be determined by the following rules: (i) when a player reaches the target location, then both win the game; (ii) if a player moves out of the boundary of the board, then both lose; (iii) if a player fails into an infinite loop, then both lose. The players need to be careful to do not kill the other by defining detrimental games rules which force the other to move out of the boundary or repeatedly go left-right or up-down in tiles of certain colours.

Collaboration in this game is not as easy as we thought. Out of 4 playtests by human players, only one ended with a win, two ended with an "infinite loop" and one ended with the "death" of a player. Often, one player puts his teammate in danger when trying hard to approach the target location. The attraction of winning the game by reaching the target alone certainly makes the players act in a selfish way and forget about the situation of their teammates. Two examples are given as follows.

Example 1: Considering Figure 10 and the fact that

- it's the player A to make a move, and
- the rule for green-coloured tiles is not defined yet.

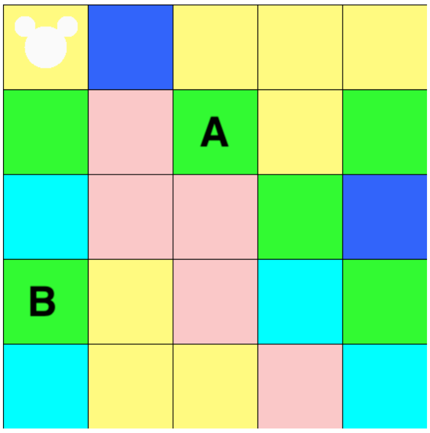


Figure 10 Player A kills player B by moving left.

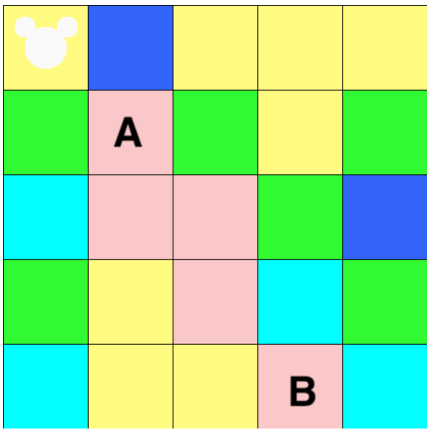


Figure 11 Player A kills player B by moving left.

If the player A moves left, then the player B will have to follow the rule by moving left and loses the game according to the criterion (ii).

Example 2: Considering Figure 11 and the fact that
- it's the player A to make a move,
- the rule for pink-coloured tiles is not defined yet, and
- the rule for yellow-coloured tile is already defined as "move to right".

If player A moves left, then the player B will have to move left and arrive at the yellow-coloured tile. The player B will fail into infinite loop and lose the game according to the criterion (iii).

Consistor with a payoff matrix

To further imitate the ethics choices and and psychology conflicts in real life, we design a more complex two-player Consistor with a payoff matrix, as given in Table 2.

Table 2 "O" refers to being out of the boundary. "I" refers to being in infinite loop. "K" refers to killing the other player, i.e., the other player is in state O or I. "M" refers to meeting each other.

$State_A$	$State_B$	$Score_A$	$Score_B$
I/O	I/O	0	0
K	I/O	1	0
M	M	3	3

Surprisingly, the same human players in the previous playtests showed better collaboration in this game than in the actual cooperative one and often ended with a double-win. The observation makes us rethink our design of cooperative games. Are the games actually designed as how we wanted? Or in other words, do the players behave as how we expect/predcit when designing the games?

Summary and Future Steps

Our initial motivation was to design a game based on consistency to imitate the human behaviour when making decisions in same situations, but we ended with some interesting observations when playing the designed two-player versions. This helps us make connections between the central ideas of the game and certain ideas from ethics, psychology, and media studies. A work is progress is using evolutionary computation to generate playable game boards of different difficulty levels and using search algorithms to solve and evaluate the designed games.

3.16 Game-Playing Agent Evaluation

Vanessa Volz (modl.ai – Copenhagen, DK), Yngvi Björnsson (Reykjavik University, IS), Michael Buro (University of Alberta – Edmonton, CA), Raluca D. Gaina (Queen Mary University of London, GB), Günter Rudolph (TU Dortmund, DE), Nathan Sturtevant (University of Alberta – Edmonton, CA), and Georgios N. Yannakakis (University of Malta – Msida, MT)

The goal of this working group was to analyse and improve how game-playing agents are evaluated. While various competitions (GVGAI, pathfinding) and milestone-benchmarks (e.g. Chess, StarCraft II) exist, it still remains difficult to draw general conclusions about the strengths and weaknesses of a given algorithm from existing results. This is mainly due to the lack of comparability between different ways of evaluating an algorithm.

The same issues make reporting on the performance of game-playing agents in publications difficult. Evaluation results are (1) likely not generalisable to other games / other domains and (2) it is often difficult to find comparable results from other publications.

As a first step towards alleviating these issues, we developed a general framework that describes several aspects of evaluating game-playing agents with the intention to define a standard of best-practices. If adopted widely, this could facilitate comparisons of different algorithms and at the same time ensure good scientific practise.

The main output of this working group was a list of aspects under which algorithms can be compared, grouped into different dimensions. In lieu of presenting the whole list, we give an overview of these dimensions below. The full list, along with 2 example applications can be found in the supplementary material.

1. Objective Measure: To compare AI approaches, the goal they are trying to achieve should be clearly stated. In-game score or winrate are the most common objectives in current research, but more subjective aspects, such as e.g. believability, play an important role as well.
2. Constraint Measures: This describes under which constraints a comparable AI should be operating. This includes e.g. computational, model and implementation complexity, as well as explainability and human knowledge engineering.
3. Problem Definition: This describes the environment in which the agent is operating. It includes the interface to the game as well as the type and number of different settings the agent is tested in.
4. Scientific Method: This relates to general best-practices in reporting empirical results, including reproducibility, hypothesis-driven analysis and self-reporting.

Our list of dimensions is hopefully a starting point to more discussions regarding meaningful and scientifically sound evaluations of game-playing agents. We plan to develop this further into more concrete guidelines in the future.

3.17 Search in Video Games

Mark Winands (Maastricht University, NL), Dan Ashlock (University of Guelph, CA), Michael Buro (University of Alberta – Edmonton, CA), and Chiara F. Sironi (Maastricht University, NL)

To make sound decisions in video games, players also have to plan ahead, just as they have to do for classic board games such as chess and checkers. This is not trivial, as it is not always obvious which actions can be taken to accomplish a goal in an effective way. For an example, for a real-time strategy game a scenario could be that agents first have to cut some trees in order to reach a mining location much quicker.

Video-games can have several properties that make search much harder than in abstract games. First, they have the real-time property, there are no turns, and players can make a decision at any time in a continuously changing world. Next, there can be an infinite number of actions, which also can be durative. These games have also a high degree of non-determinism, i.e., an action could have several outcomes. They are also partially observable, players cannot see the complete map. There is also incomplete information, the exact rules and underlying structure of the game is not known in advance. Rewards could therefore be inaccurate or even deceptive. This impact could be even stronger as rewards are sparse. Finally, there are even multiple agents (so-called NPCs) for which a player can make separate decisions at the same time.

The challenge is to abstract the search space of such a complex video game into an extensive form game, which makes game-tree search algorithms feasible. The amount of detail left out is dependent on the situation, and an abstraction mechanism should be therefore adaptive.

Abstraction can be twofold. 1) They either abstract action sequences into so-called macro-actions. Search techniques investigate these macro-actions in order to find a rational strategy. Scripts will then execute these macro-actions by making the low-level decisions, so called micro-actions. 2) They abstract the states as well. For example, by transforming the game levels to a tile-based graph. The associated actions in these abstracted worlds are high-level as well. For instance, movement can be a macro-action by going from one tile to another, where the micro-actions would handle the obstacles in the tile themselves.

In order to be able to search, the macro-actions should have access to the forward model. As the forward model is used to simulate the micro-actions, it should run faster than the game speed. The built-in forward models of many video games are rather slow, making effective search daunting. The solution is to learn a fast forward model. The approach would be as follows. A set of macro-actions is constructed, and their outcomes when applying it on the existing forward model are recorded. The next step is to learn to predict the outcome of a particular macro-action for a particular state. The resulting learned model will replace the existing forward model, which will speed up the search, as there are no micro-actions to be simulated anymore. An AlphaZero approach could be used to learn these mappings.

3.18 Forward Game Design Models

Georgios N. Yannakakis (University of Malta – Msida, MT), Alex J. Champandard (nucl.ai – Wien, AT), Michael Cook (Queen Mary University of London, GB), Mark J. Nelson (American University – Washington, US), and Julian Togelius (New York University, US)

Forward models underpin many approaches to general game playing, allowing agents to reason about the future state of the game and perform more efficient search. In this working group we reframed the way forward models are traditionally conceived and instead utilised the notion of a forward model for game design. A model that reveals the underlying processes of game design and predicts the game design state when a particular game design action is taken would allow computational game designers to predict the effect of changes to a game design. Such predictions can, in turn, assist in the automatic generation of entirely new games that adhere to particular game design goals and styles. Capturing the process of game design also offers an autonomous way for unfolding aspects of player behavior and experience as both are naturally embedded within that process. In this working group we explored how one might construct such a forward model, and suggested experimental setups that would allow for data on game design decisions to be gathered.

4 Panel discussions

4.1 Closing Session

Pieter Spronck (Tilburg University, NL)

During the closing session, the participants evaluated the seminar, reflecting on what did and did not work, and looking forward on how the seminar series should continue. The topics below were discussed.

The opening session of the week contained a “game” in which each participant would tell who they are, and then would give a true/false statement about themselves, for which they had to achieve the “biggest possible entropy” (i.e., the difference between number of people who think the statement is true and number of people who think the statement is false being as small as possible). This went fast and was still sufficiently memorable to be a good introduction, and also worked as an ice-breaker.

On the first day we asked participants to anonymously (so that junior participants would not feel intimidated by senior ones) provide a title for a working group, and then wrote the working groups on the blackboard. After that we reduced the number of working groups to an amount for which each group would have about five people. Working groups which were not held were kept on the blackboard, and would either be held later in the week, or canceled by the person who proposed them, and new working groups could be proposed during the week. This approach worked well.

Every morning of the seminar we started with brief talks, usually by participants who wanted to start a working group on a particular topic. This worked well. While the seminar is focused on working groups, and is considered successful in particular because of this, the longer talks in the evenings, which were considered optional and usually visited by about half the participants, were really appreciated. These longer talks (which could take the form of tutorials) should be incorporated explicitly for a follow-up seminar, so that more preparation is possible. This would provide a good mix of working groups and optional longer talks and plenary discussions.

There was some criticism of the intensity of the program. Some participants felt that more "moments of rest" should be included. For instance, it wouldn't be a bad idea to have a walk after lunch almost every day. The practice of taking breaks should be normalized.

There was some criticism on the "diversity" of the participants, though the organizers felt that we actually were not doing too badly at that. However, the suggestion that more (senior) people could be involved in suggesting names for invitees was taken to heart.

There was some criticism on the fact that there was a strong sense of community among a large number of participants, and that those who came from 'outside' this community sometimes felt intimidated. The organization should make more attempts to stimulate those 'outsiders' to talk. It was recognized that there definitely had been made such attempts, it still seems that more attention should be given to this issue. One possibility is to offer every participant the possibility to get a 'mentor' or 'buddy' before the seminar starts, who can discuss with them how they should approach the seminar and what is expected of them. This could also take the form of a list of 'volunteers' who can be contacted to provide such information. In this respect it might be a good idea to make someone reach out privately to potential participants before they receive the invitation, so that they know what the seminar is about and why it should not be missed.

It was suggested that because the participants of the seminar tend to be high-ranked senior researchers and promising junior researchers, due to the style of seminar (mostly working groups) it is the ideal event in which some of the future developments in the field can be determined. It would be a good idea to explicitly take this position early in the week, preferably during the opening session, by discussing "what is new right now and what will the future be like."

A list of potential topics for a follow-up seminar was discussed, among which: (1) bringing back lost topics; (2) 3+ player interaction; (3) human-game AI cooperation and interaction; (4) human-game interfaces; (5) using languages in games; and (6) game evaluation and metrics. Some of these topics (in particular the last two) may be more suitable for a small seminar.

Finally, potential relaxation activities for a future seminar were discussed, among which: (1) playing a MOBA; (2) exploring local food; (3) role-playing games; (4) mini-LARP; and (5) programming simple machines such as Arduino and Makey-makey.

The closing session took about 90 minutes, which is quite long, but the organizers found it stimulating. It showed how engaged the participants were with the seminar, and how interested they were in making a follow-up seminar even better.

Participants

- Dan Ashlock
University of Guelph, CA
- Yngvi Björnsson
Reykjavik University, IS
- Alan Blair
UNSW – Sydney, AU
- Bruno Bouzy
Nukkai – Paris, FR
- Cameron Browne
Maastricht University, NL
- Michael Buro
University of Alberta – Edmonton, CA
- Duygu Cakmak
Creative Assembly – Horsham, GB
- Tristan Cazanave
University Paris-Dauphine, FR
- Alex J. Champandard
nucl.ai – Wien, AT
- Michael Cook
Queen Mary University of London, GB
- Peter I. Cowling
University of York, GB
- Jakob Foerster
University of Toronto, CA
- Raluca D. Gaina
Queen Mary University of London, GB
- Katja Hofmann
Microsoft Research – Cambridge, GB
- Jialin Liu
Southern Univ. of Science and Technology – Shenzen, CN
- Setareh Maghsudi
TU Berlin, DE
- Nantas Nardelli
University of Oxford, GB
- Mark J. Nelson
American University – Washington, US
- Diego Perez Liebana
Queen Mary University of London, GB
- Mike Preuß
Leiden University, NL
- Sebastian Risi
IT University of Copenhagen, DK
- Günter Rudolph
TU Dortmund, DE
- Tómas Philip Rúnarsson
University of Iceland – Reykjavik, IS
- Christoph Salge
New York University, US
- Spyridon Samothrakis
University of Essex – Colchester, GB
- Tom Schaul
Google DeepMind – London, GB
- Jacob Schrum
Southwestern University – Georgetown, US
- Emily Short
Spirit AI – Oxford, GB
- Chiara F. Sironi
Maastricht University, NL
- Pieter Spronck
Tilburg University, NL
- Nathan Sturtevant
University of Alberta – Edmonton, CA
- Olivier Teytaud
Facebook – Paris, FR
- Tommy Thompson
AI and Games – London, GB
- Julian Togelius
New York University, US
- Vanessa Volz
modl.ai – Copenhagen, DK
- Hui Wang
Leiden University, NL
- Mark Winands
Maastricht University, NL
- Georgios N. Yannakakis
University of Malta – Msida, MT

Report from Dagstuhl Seminar 19512

Interactive Design and Simulation

Edited by

Thomas A. Grandine[1], Jörg Peters[2], and Ulrich Reif[3]

1 **The Boeing Company – Seattle, US,** thomas.a.grandine@boeing.com
2 **University of Florida – Gainesville, US,** jorg.peters@gmail.com
3 **TU Darmstadt, DE,** reif@mathematik.tu-darmstadt.de

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 19512 “"Interactive Design and Simulation”. After the executive summary, the collection of abstracts of the presentations forms the core of this report, complemented by an example of working group results that highlights the diversity of backgrounds and approaches.

Seminar December 15–20, 2019 – http://www.dagstuhl.de/19512

2012 ACM Subject Classification Computing methodologies → Computer graphics, Computing methodologies → Computer vision, Computing methodologies → Modeling and simulation, Mathematics of computing → Partial differential equations, Computing methodologies → Real-time simulation, Mathematics of computing → Spline models

Keywords and phrases simulation of physical systems, geometric models for engineering analysis, partial differential equations, interactive and real-time computation splines, model reduction

Digital Object Identifier 10.4230/DagRep.9.12.115

Edited in cooperation with Angelos Mantzaflaris

1 Executive Summary

Jorg Peters (University of FLorida, USA)

Dagstuhl Seminar 19512 presented and debated a rich set of techniques for improving algorithms and interfaces for interactive physical simulation, based on geometric and physical models and governed by partial differential equations. The techniques originate in geometry processing, computational geometry, geometric design, and the use of splines in meshing-less and iso-geometry approaches.

Thanks to its diverse roster of participants, with expertise spanning computer science, applied mathematics and engineering, the seminar enabled rare new interactions between academia, industrial and government-sponsored labs and fostered new insights apart from technical considerations. For example, one of the ad hoc discussions centered around the mechanisms and person-to-person considerations that enable transfer of new techniques from academia to industry. Another discussion focused on bridging the divide between geometric modeling and engineering analysis. A third focused on the usage (or lack thereof) of academic open-source libraries. And a fourth elucidated the different error measures that allow or prevent model reduction techniques for non-linear models (e.g. of elasticity) for given applications ranging from animation to product design.

The seminar was well-paced, avoiding densely-packed presentations. An emphasis was placed on time to formulate both specific and long range challenges. The benchmark problem in Section 4 is an example of specific problems that clarify and contrast the competing approaches and objectives and advertised the different strengths and the synergy of the areas: responses that permit two-sided error bounds, responses based on mathematical reformulation, applying advanced computational geometry and new software packages that leverage hierarchical spline software.

2 Table of Contents

3 Overview of Talks

3.1 Animation and Simulation of Realistic Virtual Humans

Mario Botsch (Universität Bielefeld, DE)

Joint work of Jascha Achenbach, Mario Botsch, Dominik Gall, Martin Komaritzan, Marc Latoschik, Daniel Roth, Thomas Waltemate
Main reference Martin Komaritzan, Mario Botsch: "Projective Skinning", Proc. ACM Comput. Graph. Interact. Tech., Vol. 1(1), The Association for Computers in Mathematics and Science Teaching, 2018.
URL http://dx.doi.org/10.1145/3203203

Realistic virtual humans have many interesting applications in several fields–ranging from characters in computer games, to avatars representing ourselves in virtual environments, and digital models for medical diagnosis and intervention planning.

Over the last five years we built a fast avatar generation pipeline that produces an accurate virtual clone of a real person in less than 10 minutes [2]. We first scan the real person by taking photos in a multi-camera rig consisting of 48 synchronized DSLR cameras. A multi-view photogrammetric reconstruction computes a dense point cloud from these images. In order to robustly cope with noise, outliers, and missing data we fit a statistical human body template to the high-resolution point cloud, producing an accurate geometric reconstruction. An anisotropic discrete curvature model allows to faithfully reconstruct folds and wrinkles even in the presence of noise [1]. A high-resolution texture is finally generated from the reconstructed geometry and the input camera images. The resulting avatars were evaluated in a virtual mirror scenario and were shown in significantly improve perceived body ownership and perceived presence [3].

Many real-time character animation approaches, such as Linear Blend Skinning or Dual Quaternion Skinning, are known to produce artifacts in the vicinity of strongly bent joints. Our *Projective Skinning* yields physically plausible animations by combining an articulated embedded volumetric skeleton with a real-time projective-dynamics simulation of the elastic tissue enclosed between skin and bone [4]. This approach has recently been accelerated through a custom sparse matrix format, a GPU-based conjugate gradients solver, and an MLS-based quadratic upsampling technique [5]. The resulting *Fast Projective Skinning* features plausible tissue deformation, dynamic motion effects (jiggling), as well as global collision handling–while still being fast enough for real-time applications.

References

1 Jascha Achenbach, Eduard Zell, Mario Botsch. *Accurate Face Reconstruction through Anisotropic Fitting and Eye Correction*. Proceedings of VMV 2015.
2 Jascha Achenbach, Thomas Waltemate, Marc Latoschik, Mario Botsch. *Fast Generation of Realistic Virtual Humans*. Proceedings of ACM VRST 2017.
3 Thomas Waltemate, Dominik Gall, Daniel Roth, Mario Botsch, Marc Erich Latoschik. *The Impact of Avatar Personalization and Immersion on Virtual Body Ownership, Presence, and Emotional Response*. IEEE TVCG 24(4), 2018.
4 Martin Komaritzan, Mario Botsch. *Projective Skinning*. Proceedings of ACM i3D 2018.
5 Martin Komaritzan, Mario Botsch. *Fast Projective Skinning*. Proceedings of ACM MIG 2019

3.2 Mollified finite element approximants of arbitrary order and smoothness

Fehmi Cirak (University of Cambridge, GB)

Main reference Eky Febrianto, Michael Ortiz, Fehmi Cirak: "Mollified finite element approximants of arbitrary order and smoothness", CoRR, Vol. abs/1910.04002, 2019.
URL https://arxiv.org/abs/1910.04002

The approximation properties of the finite element method can often be substantially improved by choosing smooth high-order basis functions. It is extremely difficult to devise such basis functions for partitions consisting of arbitrarily shaped polytopes. We propose the mollified basis functions of arbitrary order and smoothness for partitions consisting of convex polytopes. On each polytope an independent local polynomial approximant of arbitrary order is assumed. The basis functions are defined as the convolutions of the local approximants with a mollifier. The mollifier is chosen to be smooth, to have a compact support and a unit volume. The approximation properties of the obtained basis functions are governed by the local polynomial approximation order and mollifier smoothness. The convolution integrals are evaluated numerically first by computing the boolean intersection between the mollifier and the polytope and then applying the divergence theorem to reduce the dimension of the integrals. The support of a basis function is given as the Minkowski sum of the respective polytope and the mollifier. The breakpoints of the basis functions, i.e. locations with non-infinite smoothness, are not necessarily aligned with polytope boundaries. Furthermore, the basis functions are not boundary interpolating so that we apply boundary conditions with the non-symmetric Nitsche method as in immersed/embedded finite elements. The presented numerical examples confirm the optimal convergence of the proposed approximation scheme for Poisson and elasticity problems.

3.3 Adaptive isogeometric methods with (T)HB-splines: suitably graded meshes and BPX preconditioners

Carlotta Giannelli (University of Firenze, IT)

Joint work of Carlotta Giannelli, Cesare Bracco, Durkbin Cho, Rafael Vázquez

We consider adaptive isogeometric methods with (truncated) hierarchical B-splines, usually denoted as (T)HB-splines. In virtue of a simple multilevel construction, the hierarchical spline model represents one of the most successful choice for performing adaptivity in the framework of isogeometric analysis. In this talk I will present recent results towards the efficient use of (T)HB-splines. In particular, the construction of an additive multilevel preconditioner, also known as BPX preconditioner, will be discussed. By exploiting the locality of hierarchical spline functions, efficient multilevel decompositions can be identified to prove that, when suitably graded hierarchical meshes are considered, the condition number of the preconditioned system is bounded independently of the number of levels. To highlight the performance of the preconditioner, the theoretical results will be presented together with a selection of numerical examples.

3.4 Grasping and manipulation: Representation of shape and dynamics

Cindy Marie Grimm (Oregon State University – Corvallis, US)

Robotic manipulation is a complex combination of four different elements: 1) The kinematics of the manipulator, 2) The sensors, 3) The actuators, and 4) the control strategy that maps sensors to actuators. In order to optimize in this space, i.e., successfully complete a manipulation task, we have to optimize over all 4 elements. To make this manageable, we propose a general approach that allows us to optimize element-wise. Specifically, we use humans as ideal actuators and controllers, freeing us to explore the kinematic structure. Essentially, we ask humans to "puppeteer" the robotic manipulators through structured tasks. By instrumenting the robotic hands and the objects that are manipulated, we can build up data on how well specific kinematic structures perform. Simultaneously, we also capture actuation, sensor, and control information. Given this, we can now systematically add in actuators and controllers.

Within this learning structure, there are still several open-ended questions that require both shape and dynamic models. Specifically, what are effective models for representing shape and dynamics that are amenable to real-world sensors? We also need statistically accurate physical simulators that allow us to optimize designs for manipulation. Typical manipulators consist of compliant materials connected with both rigid body and soft joints, with cable actuators. Contact is extremely important; and it is important to not only model contact but model it in a statistical manner – how might an object be moved as a result of the contact, not just a single sample.

3.5 Real-Time Simulation Of Elastic Solids and Fluid-Solid Interaction

Klaus Hildebrandt (TU Delft, NL)

Joint work of Klaus Hildebrandt, Christopher Brandt, Leonardo Scandolo, Elmar Eisemann
Main reference Christopher Brandt, Leonardo Scandolo, Elmar Eisemann, Klaus Hildebrandt: "The reduced immersed method for real-time fluid-elastic solid interaction and contact simulation", ACM Trans. Graph., Vol. 38(6), pp. 191:1–191:16, 2019.
URL http://dx.doi.org/10.1145/3355089.3356496
Main reference Christopher Brandt, Elmar Eisemann, Klaus Hildebrandt: "Hyper-reduced projective dynamics", ACM Trans. Graph., Vol. 37(4), pp. 80:1–80:13, 2018.
URL http://dx.doi.org/10.1145/3197517.3201387

This talk splits in two parts. In the first part, we discuss Hyper-Reduced Projective Dynamics (HRPD) an approach for the real-time simulation of deformable objects that combines the robustness, generality, and high performance of Projective Dynamics with the efficiency and scalability offered by model reduction techniques. The method decouples the cost for time integration from the mesh resolution and can simulate large meshes in real-time. In the second part, we look at a method for the real-time simulation of two-way coupled incompressible fluids and elastic solids called the Reduced Immersed Method (RIM). It is based on a novel discretization of the immersed boundary equations of motion that combines HRPD with a PIC/FLIP fluid solver. Crucial for the performance of RIM is the efficient transfer of information between the elasticity and the fluid solver and the synchronization of the Lagrangian and Eulerian settings. We introduce the concept of twin subspaces that enables an efficient reduced-order modeling of the transfer.

3.6 Adaptive spline projectors via restricted hierarchical spline fitting

Bert Jüttler (Johannes Kepler Universität Linz, AT)

Joint work of Bert Jüttler, Alessandro Giust, Angelos Mantzaflaris

This talk is devoted to techniques for adaptive spline projection via quasi-interpolation, enabling the efficient approximation of given sample data or functions. We employ local least-squares fitting in restricted hierarchical spline spaces to establish novel projection operators for hierarchical splines of degree p. This leads to efficient spline projectors that require $\mathcal{O}(p^d)$ floating point operations and $\mathcal{O}(1)$ evaluations of the given function per degree of freedom, while providing essentially the same accuracy as global approximation. The spline projectors, which are based on a unifying framework for quasi-interpolation in hierarchical spline spaces, are shown to compare favorably with other constructions that have been described in the rich literature on this topic.

3.7 Computational Modeling for Cardiac Biomechanics

Adarsh Krishnamurthy (Iowa State University, US)

Joint work of Adarsh Krishnamurthy, Arian Jafari, Edward Pszczolkowski, Aditya Balu, Ming-Chen Hsu, Soumik Sarkar
Main reference Arian Jafari, Edward Pszczolkowski, Adarsh Krishnamurthy: "A framework for biomechanics simulations using four-chamber cardiac models", Journal of Biomechanics, Vol. 91, pp. 92–101, 2019.
URL https://doi.org/10.1016/j.jbiomech.2019.05.019

Cardiovascular diseases, such as heart failure, are one of the leading causes of death in the world and pose a severe burden to the healthcare system. Computational models of the cardiovascular system, developed from patient-specific clinical data, can help refine the diagnosis and personalize the treatment. We have been working on developing an integrative framework for cardiac biomechanics with simulation, analysis, and visualization tools that will significantly advance the state-of-the-art in personalized medicine. In this talk, I presented recent advances in computational modeling that enables simulation of a four-chamber cardiac model [1]. We have developed tools to generate a patient-specific cubic-Hermite four-chamber cardiac mesh and use isogeometric analysis methods to simulate a full cardiac cycle [2].

The second part of the talk focused on novel machine-learning algorithms to optimize the design of bioprosthetic heart valves. Machine-learning tools can significantly accelerate biomechanics simulations, leading to the development of a high-fidelity surrogate model, which can then be used for optimizing the geometry of valves. This surrogate model can then be used for optimizing the geometry of bioprosthetic valves, leading to patient-specific valves with better fit and performance, reducing the need for premature valve replacements [3]. The tools and methods developed as part of this research will help improve patient care and treatment outcomes, ultimately benefiting society with improved healthcare.

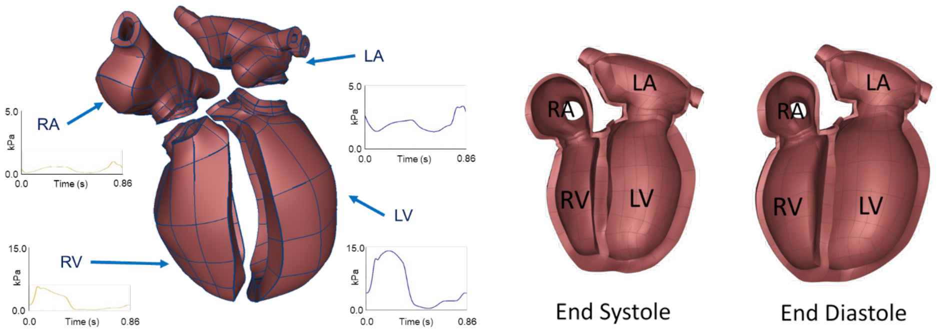


Figure 1 Four chamber cardiac model. The cavity pressures are applied to the inner surfaces of the four chambers to simulate a complete cardiac cycle. Two deformed geometries at end systole and end diastole are shown on the right.

References

1 Arian Jafari, Edward Pszczolkowski, and Adarsh Krishnamurthy; A framework for biomechanics simulations using four-chamber cardiac models; *Journal of Biomechanics*, **91**:92-101, 2019.

2 Adarsh Krishnamurthy, Matthew Gonzales, Gregory Sturgeon, Paul Segars, and Andrew McCulloch; Biomechanics simulations using cubic-Hermite meshes with extraordinary nodes for isogeometric cardiac modeling," *Computer-Aided Geometric Design*, **43**:27-38, 2016.

3 Aditya Balu, Sahiti Nallagonda, Fei Xu, Adarsh Krishnamurthy, Ming-Chen Hsu, and Soumik Sarkar; A deep learning framework for diagnostics and patient-specific design of bioprosthetic heart valves; *Scientific Reports*, **9**, 2019.

3.8 Robust Cut-Cells for Triangle Meshes

David I. W. Levin (University of Toronto, CA)

Joint work of David I. W. Levin, Michael Tao, Christopher Batty, Eugene Fiume
Main reference Michael Tao, Christopher Batty, Eugene Fiume, David I. W. Levin: "Mandoline: robust cut-cell generation for arbitrary triangle meshes", ACM Trans. Graph., Vol. 38(6), pp. 179:1–179:17, 2019.
URL https://doi.org/10.1145/3355089.3356543

Although geometry arising "in the wild" most often comes in the form of a surface representation, a plethora of geometrical and physical applications require the construction of volumetric embeddings either of the geometry itself or the domain surrounding it. Cartesian cut-cell-based mesh generation provides an attractive solution in which volumetric elements are constructed from the intersection of the input surface geometry with a uniform or adaptive hexahedral grid. This choice, especially common in computational fluid dynamics, has the potential to efficiently generate accurate, surface-conforming cells; unfortunately, current solutions are often slow, fragile, or cannot handle many common topological situations. We therefore propose a novel, robust cut-cell construction technique for triangle surface meshes that explicitly computes the precise geometry of the intersection cells, even on meshes that

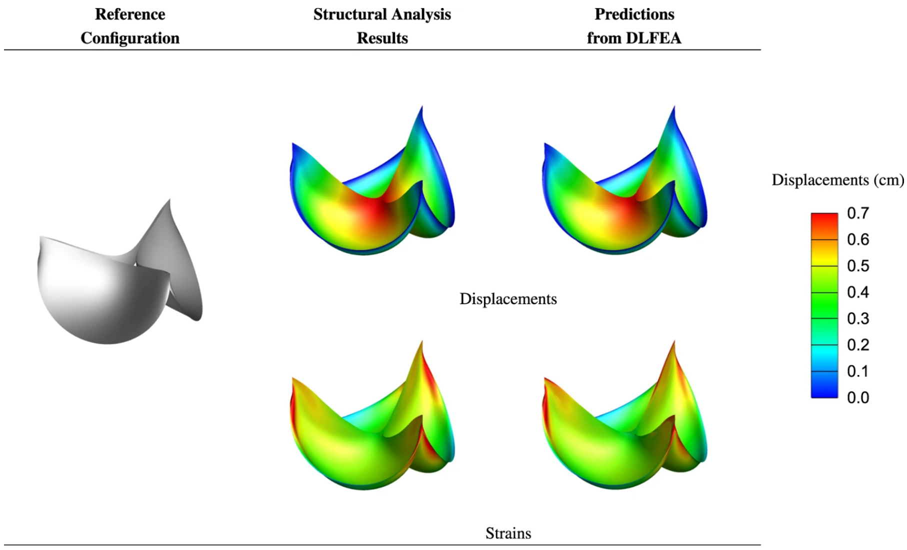


Figure 2 Illustrative example of valve deformations and their corresponding maximum in-plane principal Green-Lagrange strains computed from isogeometric simulations and predicted using deep learning.

are open or non-manifold. Its fundamental geometric primitive is the intersection of an arbitrary segment with an axis-aligned plane. Beginning from the set of intersection points between triangle mesh edges and grid planes, our bottom-up approach robustly determines cut-edges, cut-faces, and finally cut-cells, in a manner designed to guarantee topological correctness. We demonstrate its effectiveness and speed on a wide range of input meshes and grid resolutions, and make the code available as open source.

3.9 Fast algorithms for hierarchical bases over tensor-product splines

Angelos Mantzaflaris (INRIA – Valbonne, FR)

In this talk we present methods to improve the efficiency of computations with hierarchical splines that are constructed over tensor-product spline spaces. We focus on the problem of computing the Gram matrix of the basis. Typically, computations involve numerical integration using tensor-product Gauss quadrature. However, it is known that an element-wise assembly of the Gramian of tensor product B-splines is sub-optimal in dimension bigger than one. We present efficient algorithms for this computation using the background tensor structure. In particular, we extend the Kronecker formula that is known for the Gramian of tensor-product spaces to a Hadamard formula for the Gramian of hierarchical bases. This implies an efficient algorithm for the computation of the matrix, that does not involve a multivariate quadrature over the elements.

3.10 Current challenges in industrial surface reconstruction at MTU

Dominik Mokriš (MTU Aero Engines – München, DE)

Joint work of Dominik Mokriš, Johannes Barner, David Großmann, Urška Zore

The aim of the talk is to contribute to the exchange of ideas and experience between academia and industry. This is achieved both by explaining some of the specifics of implementing geometric algorithms in industrial setting and by showing several hand-picked problems where we expect the attending scientists to be able to suggest effective remedies.

In the first part, I give a broad idea of how our geometry generator (the purpose and operation of which are covered in the talk of Urška Zore) is organised and what are its interfaces. Although seemingly irrelevant to the scientific contents, such interfaces are crucial in the industry, as they reflect how the tasks are distributed between various specialists. The ability of a newly proposed method to conform to pre-existing interfaces can be one of the deciding factors of its acceptance at a particular company.

To demonstrate the principle I show several examples of how we use the scientific libraries G+Smo and CGAL that are well-known and co-developed by several of the participants.

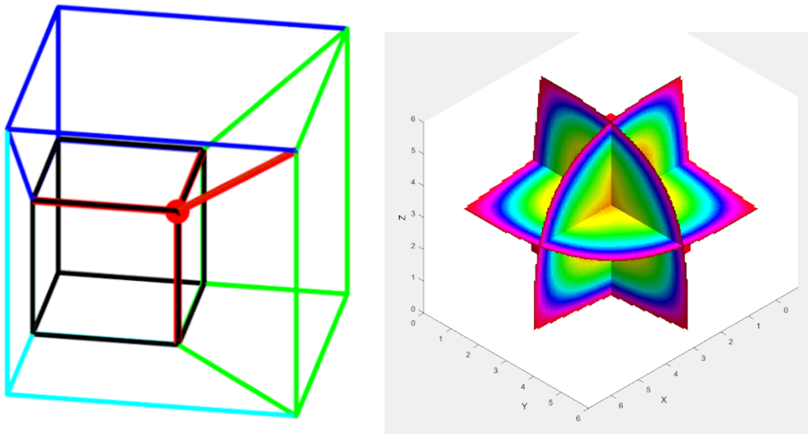
In the final part, I concentrate on explaining several practical geometric problems for which the other participants might have suggestions. These problems come mainly from the area of surface reconstruction (which despite its already big number of practial applications keeps finding new ones), where improvements in surface fitting and (re-)parametrisations of triangulations allow for higher level of precision without introducing artifacts. Another briefly mentioned problem is the possibility to use isogeometric analysis (or other PDE-related methods) in geometric modeling.

3.11 Refinable tri-variate C^1 splines for box-complexes including irregular points and irregular edges

Jörg Peters (University of Florida – Gainesville, US)

The grid points of a regular partition of 3-space into boxes can be interpreted as the control points of a tri-variate tensor-product spline with one polynomial piece per cube. The theory of such splines is well-understood. By contrast, for box-complexes where the tensor-grid gives way to an irregular arrangement of boxes including irregular points and irregular edges, there is to date no simple prescription to join the corresponding polynomial pieces with more than C^0 continuity. Efficiently modeling C^1 fields over general box-complexes is of interest in areas ranging from scientific data visualization to solving higher-order differential equations. For example, to visualize a flow computed by the Discontinuous Galerkin approach currently requires substantial post-processing to extract stream lines that the theory predicts to be smooth.

Already in two variables – where the box-complex is a quad mesh and the only irregularities are points where more or fewer than four quadrilaterals meet – associating one or more bi-cubic polynomial pieces with each quad and joining them to form a C^1 space is far from trivial: (i) Geometric continuity requires increased polynomial degree near irregularities



(a) box-complex with irregular point ($n = 4$) and irregular edges ($n_e = 3$) **(b)** Solution of $\Delta(u \circ \mathbf{x}) = 1$

Figure 3 Modeling and computing with refinable tri-variate C^1 splines for box-complexes including (a) four irregular edges of valence 3 and one irregular vertex of valence 4. (b) The four domains map to curved boxes partitioning an octant of a ball and Poisson's equation is solved on the octant, by collocation.

and careful book-keeping to adjust reparameterizations under refinement; (ii) Subdivision creates an infinite sequence of nested piecewise polynomial rings that complicate engineering analysis, e.g. integration, near irregularities; and (iii) vertex-singular parameterizations can have poor shape deficient and must ensure that the singularity is locally removable.

In three variables, tri-variate Catmull-Clark subdivision lacks a guarantee of smoothness and approximation order; and geometric continuity, although well-understood in principle, is in practice barely explored.

This talk introduced a trivariate C^1 space with singular parameterization. Wherever possible, the vertices of the box-complex are interpreted as B-spline coefficients. Then, at each irregularity, a well-behaved linear function is determined and composed with a local singular expansion $\mathbf{x}$ that is based on the intersection of edge-dual planes within each box. All first derivatives of the expansion $\mathbf{x}$ are continuous, albeit zero across irregularities. Apart from the irregularities, its Jacobian is positive definite so that $\mathbf{x}^{-1}$ is well defined. Evaluating the local expansion of the linear function composed with $\mathbf{x}$ at $\mathbf{x}^{-1}$ removes the singularity. The polynomial pieces of the spline space therefore join not just nominally C^1, but smoothly over the whole box-complex. The spline space has a basis of $2 \times 2 \times 2$ independent functions per hexahedral input box (one per sub-box after a dyadic split in each dimension), can reproduce linear functions and is refinable.

Fig. 3a shows a box complex with irregular point ($n = 4$) and four irregular edges ($n_e = 3$). The corresponding piecewise tri-cubic map is smooth across the irregularities and parameterizes an octant of a ball. To test the construction as physical domain, the Poisson equation is solved on the octant, with zero boundary conditions. Fig. 3b shows slices colored by the resulting scalar field satisfying the Poisson equation in the sense of collocation.

3.12 From Delaunay Triangulations to Curved Optimal Delaunay Triangulations

Pierre Alliez (INRIA – Valbonne, FR)

Joint work of Leman Fen, Laurent Busé, Hervé Delingette, Mathieu Desbrun, Pierre Alliez
Main reference Leman Feng, Pierre Alliez, Laurent Busé, Hervé Delingette, Mathieu Desbrun: "Curved optimal delaunay triangulation", ACM Trans. Graph., Vol. 37(4), pp. 61:1–61:16, 2018.
URL http://dx.doi.org/10.1145/3197517.3201358

Meshes with curvilinear elements hold the appealing promise of enhanced geometric flexibility and higher-order numerical accuracy compared to their commonly-used straight-edge counterparts. However, the generation of curved meshes remains a computationally expensive endeavor with current meshing approaches: high-order parametric elements are notoriously difficult to conform to a given boundary geometry, and enforcing a smooth and non-degenerate Jacobian everywhere brings additional numerical difficulties to the meshing of complex domains. In this talk I will present an extension of Optimal Delaunay Triangulations (ODT) to curved and graded isotropic meshes. By exploiting a continuum mechanics interpretation of ODT instead of the usual approximation theoretical foundations, we formulate a very robust geometry and topology optimization of Bézier meshes based on a new simple functional promoting isotropic and uniform Jacobians throughout the domain. The resulting curved meshes can adapt to complex domains with high precision even for a small count of elements thanks to the added flexibility afforded by higher order basis functions.

3.13 Quasi-interpolants and the solution of fractional differential problems

Francesca Pitolli (Sapienza University of Rome, IT)

Nowadays, Fractional Calculus is a well-established tool to model a great variety of real-world phenomena, from viscoelasticity to population growth, from anomalous diffusion to wave propagation. The growing popularity of differential problems having derivatives of fractional, i.e. noninteger, order is due to their ability to model nonlocality in space or memory effects in time.

Unfortunately, the analytical solution of fractional differential problems is known just in some special cases and it is often expressed as a series expansion. Thus, there is a great effort in constructing efficient numerical methods to solve this kind of problems.

To this end, collocation methods have proved to be particularly effective since their are global methods that can easily take into account the nonlocal behavior of the fractional derivative. Here, we construct a collocation method based on spline quasi-interpolants and use it to solve boundary differential problems having space derivative of fractional order. The main advantage of this method is that the fractional derivative of the spline basis functions can be evaluated explicitly by a differentiation rule that involves the finite difference operator. We analyze the approximation properties of the method and show some numerical results.

3.14 A Few Thoughts about Design and Simulation

Ulrich Reif (TU Darmstadt, DE)

Trimmed NURBS are the standard of industrial surface modeling, today. This talk addresses two main challenges in this context: First, composite trimmed NURBS surfaces typically reveal gaps along boundary curves. Second, simulation on domains bounded by trimmed NURBS surfaces requires a time-consuming and sometimes very complicated meshing process. The latter statement is true even for Isogeometric Analysis, which requests a volumetric parametrization of the domain, which cannot be derived easily from a given boundary representation.

We suggest a new class of trimmed NURBS surfaces with accurate boundary control, called ABC-surfaces, which provably solves the first problem and offers a promising new idea to deal with the second one. Given a tensor product spline surface b describing the interior shape of the desired surface, called the base, and auxiliary spline surfaces r_ℓ representing the shape near the boundary, called ribbons, the corresponding ABC-surface a is defined by

$$a = \frac{wb + \sum_\ell w_\ell r_\ell \circ \kappa_\ell}{w + \sum_\ell w_\ell}.$$

Here, w is a function vanishing at the boundary, and the w_ℓ are functionis vanishing at all boundary segments except for that with index ℓ. Further, κ_ℓ is a reparametrization used to adapt the ribbons to the base, characterized by $b \approx r_\ell \circ \kappa_\ell$. With this setting, the surface a inherits its shape at the boundary from the ribbons and thus can be deigned easily to match the geometry of neighboring spline surfaces. In principle, it is possible to ensure a G^k-contact for any order k.

Just as it is possible in this way to parameterize surfaces with a prescribed boundary curve, it is possible to parameterize volumes with a prescribed boundary surface. This idea, which can be regarded as a combination of web-splines and Isogeometric Analysis, might have some potential to solve the notorious meshing problem in simulation, but this line of research is still in its infancy.

3.15 Quadrature schemes based on quasi-interpolation for Isogeometric Boundary Element Methods

Maria Lucia Sampoli (University of Siena, IT)

Joint work of Maria Lucia Sampoli, Alessandra Aimi, Francesco Calabrò, Tadej Kanduč, Antonella Falini, Carlotta Giannelli Alessandra Sestini

Boundary Element Methods (BEMs) are schemes studied since the mid '80s, for the numerical solution of those Boundary Valued Problems, which can be reformulated as a system of integral equations defined only on the boundary of the domain. These methods have two main advantages, the dimension reduction of the computational domain and the simplicity for treating external problems. One of the important challenges in this topic is to accurately and efficiently solve singular integrals that arise from the boundary integral equations so formulated. Therefore, designing suitable quadrature schemes is one of the main active research topic in BEM, [1].

Recently new quasi-interpolation (QI) based quadrature rules have been introduced specifically for IgA-BEM setting, [2]. Such quadrature schemes are tailored for B-splines and provide very good accuracy and optimal convergence rate. In this talk we show how weakly, strongly and hyper-singular integrals related to the 2D integral formulation of the Laplace equation with different types of boundary conditions can be approximated by using these new rules, exhibiting promising results. Moreover local refinability of the approximated solution of the problem is also addressed by using hierarchical B-spline spaces. It can be seen that the local nature of the QI perfectly fits with hierarchical spline constructions and leads to an efficient and accurate numerical scheme, [3].

The research is part of a collaboration with A. Aimi, F. Calabrò, T. Kanduč, A. Falini, C. Giannelli and A. Sestini.

References

1 Aimi, A., Calabrò, F., Diligenti, M., Sampoli, M. L., Sangalli, G.and Sestini, A. Efficient assembly based on B-spline tailored quadrature rules for the IgA-SGBEM, *Comput. Methods Appl. Mech. Engrg.* (2018) **331**:327–342.

2 Calabrò, F., Falini, A., Sampoli, M.L. and Sestini, A. Efficient quadrature rules based on spline quasi-interpolation for application to IGA-BEMs, *Journal of Comp. Appl. Math.* (2018) **338**:153–167.

3 Falini, A., Giannelli, C., Kanduč, T., Sampoli, M.L. and Sestini, A. An adaptive IgA-BEM with hierarchical B-splines based on quasi-interpolation quadrature schemes, *Int. J. Numer. Methods Eng.* (2019) **117**(10):1038–1058.

3.16 Multi-patch discretizations for isogeometric analysis

Thomas Takacs (Johannes Kepler Universität Linz, AT)

Joint work of Thomas Takacs, Mario Kapl, Giancarlo Sangalli, Somayeh Kargaran, Bert Jüttler, Stefan K. Kleiss, Angelos Mantzaflaris

In this talk we discuss different approaches to represent geometrically difficult domains using an isogeometric discretization. Isogeometric analysis (IGA) is based on the geometry representation of CAD models, which are usually composed of (possibly trimmed) B-spline or NURBS patches. This talk is covers two topics. In the first part we discuss the construction of C^1 isogeometric spaces [1], in the second part we present a method that can handle overlapping multi-patch domains [2].

In IGA globally C^1 smooth spaces over unstructured meshes can be used to discretize and solve fourth order partial differential equations using a Galerkin approach. We focus on C^0-conforming multi-patch domains. We present the construction of a specific C^1 isogeometric spline space for the class of so-called analysis-suitable G^1 (AS-G^1) multi-patch parametrizations. We analyze the properties of the space and present an approach how to handle non-AS-G^1 parametrizations, by relaxing the smoothness criteria.

Finally we present a method to solve second order PDEs on overlapping multi-patch domains. We define a coupled problem, where the solution on every patch is given as a combination of local solutions on the patch together with contributions from the neighboring patches. The resulting system can be solved directly, in contrast to iterative methods such as multiplicative/additive Schwarz.

References

1 Kapl, Sangalli, Takacs, An isogeometric C^1 subspace on unstructured multi-patch planar domains, Computer Aided Geometric Design, 69: 55-75, 2019.

2 Kargaran, Jüttler, Kleiss, Mantzaflaris, Takacs, Overlapping multi-patch structures in isogeometric analysis, Computer Methods in Applied Mechanics and Engineering, 356: 325-353, 2019.

3.17 Wave Packets on Surfaces

Chris Wojtan (IST Austria – Klosterneuburg, AT)

Joint work of Stefan Jeschke, Camille Schreck, Christian Hafner, Tomas Skrivan, Ken Museth, Andreas Soderstrom, Christoph Sprenger, John Johansson, Matthias Mueller-Fischer, Nuttapong Chentanez, Miles Macklin, Chris Wojtan
Main reference Stefan Jeschke, Chris Wojtan: "Water wave packets", ACM Trans. Graph., Vol. 36(4), pp. 103:1–103:12, 2017.
URL https://doi.org/10.1145/3072959.3073678
Main reference Stefan Jeschke, Tomás Skrivan, Matthias Müller-Fischer, Nuttapong Chentanez, Miles Macklin, Chris Wojtan: "Water surface wavelets", ACM Trans. Graph., Vol. 37(4), pp. 94:1–94:13, 2018.
URL https://doi.org/10.1145/3197517.3201336
Main reference Camille Schreck, Christian Hafner, Chris Wojtan: "Fundamental solutions for water wave animation", ACM Trans. Graph., Vol. 38(4), pp. 130:1–130:14, 2019.
URL https://doi.org/10.1145/3306346.3323002

Computational fluid simulations can be used to solve for the motion of water surface waves. These waves are especially important in computer animations applications (real-time or off-line virtual worlds). Qualitative effects like wave reflection, refraction, diffraction, and dispersion are especially visually salient and important to capture. Unfortunately, typical simulation approaches like finite elements, finite differences, and spectral/Fourier methods either fail to capture boundary effects or provide insufficient accuracy/detail due to signal processing and frequency limitations.

We propose using wave packets, essentially a wavelet basis, for animating water surface waves. The discretization explicitly models dispersion and wave group effects, and the frequency is independent of the computational degrees of freedom. We present analytic techniques for evolving wave packets, Eulerian wavelets on a flat surface, and radially symmetric Greens functions for boundary integral problems. Finally, we introduce a new technique for seeding and evolving wave packets on moving manifolds, deriving new equations of motion depending on the surface normal, acceleration, and deformation. The technique appears useful for enhancing the visual detail of an existing 3D computational fluid dynamics simulation.

3.18 Industrial practices in geometric modelling at MTU Aero Engines

Urška Zore (MTU Aero Engines – München, DE)

Joint work of Urška Zore, David Großmann, Carlotta Giannelli, Nora Engleitner, Bert Jüttler, Dominik Mokriš, Johannes Barner
Main reference Gábor Kiss, Carlotta Giannelli, Urška Zore, Bert Jüttler, David Großmann, Johannes Barner: "Adaptive CAD model (re-)construction with THB-splines", Graphical Models, Vol. 76(5), pp. 273–288, 2014.
URL https://doi.org/10.1016/j.gmod.2014.03.017

MTU Aero Engines, as one of the leading aircraft engine manufacturers, is continuously extending and improving the state-of-the-art in engineering design, in order to increase its own long-term competitiveness. This requires highly efficient and flexible geometric methods and technologies, combining newest research results, real-world product design and repair inspection processes directly used by the specialists. MTU invested heavily into the use of adaptive splines, such as truncated hierarchical B-Splines (THB-Splines) and patchwork B-Splines (PB-Splines), as generalizations of standard tensor-product splines, in order to tackle demanding industrial practices. We show two concrete examples of our recent applications: model reconstruction of measured data for analysis as part of reverse engineering process, and lofting of airfoil profiles for flexible geometric modelling and design of crucial engine components. We show how we use these promising concepts to extend the design space for an enhanced product performance, thereby improving the geometric construction for a tight CAD-CAE-CAM integration.

4 The L-shape challenge

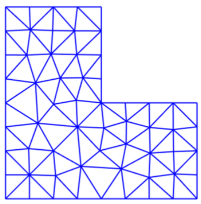


Figure 4 triangulation for classical linear finite elements.

To establish the state of the art, the working group challenged the participants to use their techniques and software packages to solve the following problem posed originally by Abinand Gopal and Lloyd N. Trefethen, see https://arxiv.org/abs/1902.00374: Compute at $(0.99, 0.99)$ the solution to the Laplace equation with zero boundary conditions on an L-shape with re-entrant corner at $(1, 1)$. The answer is known to be $u = 1.02679192610\ldots$ but is hard to compute due to the rapid change of the solution.

Earlier responses to the problem were collected in the NA Digest mailing list (Nov. 2018) and received ca 20 replies, most of which generated 2–4 correct digits, with only one close to 8 digits.

As a specific example, one response using $158,997$ degrees of freedom (DoFs) of 5th-order triangular finite elements generated 6 digits. Some of the interesting results generated by the workshop participants include:

- Least squares collocation (U. Reif) using 10,000 quadratic Hierarchical B-splines (HBs) reproduced just two digits, but unlike other approaches provides a two-sided error bound: 1.007±0.02.

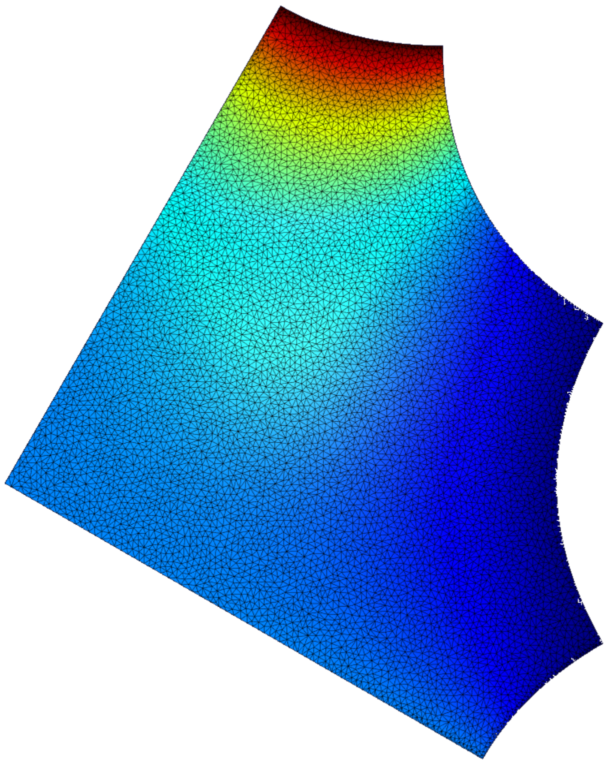


Figure 5 Conformal mapping prior to computation.

- Linear elements (cf. Fig. 4) on a uniform Delaunay mesh using the software package CGAL (P. Alliez:) and 800,000 DoFs reproduced 4 digits: 1.0264 . . . .
- Linear elements applied after *a priori* resolving the singularity at the reentrant corner via a conformal mapping, followed by uniform Delaunay triangulation with 13,699 DoFs (E. Vouga) reproduced 1.026792. . . (see Fig. 5).
- Adaptively refined HBs, ∼10 levels, deg. 5 by the software G+Smo (A. Mantzaflaris) using 30,793 DoFs reproduced 7 digits: 1.0267915. . . (see Fig. 6).

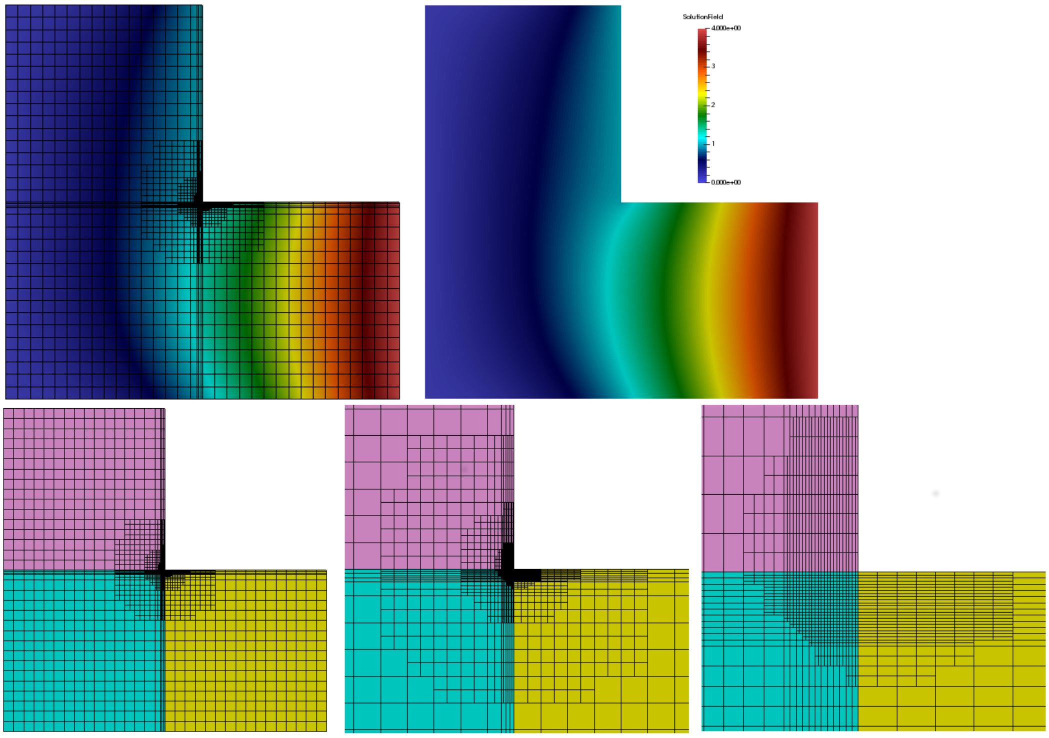


Figure 6 Hierarchical Splines.

Participants

- Pierre Alliez
INRIA – Valbonne, FR
- Mario Botsch
Universität Bielefeld, DE
- Fehmi Cirak
University of Cambridge, GB
- Carlotta Giannelli
University of Firenze, IT
- Thomas A. Grandine
The Boeing Company – Seattle, US
- Cindy Marie Grimm
Oregon State University – Corvallis, US
- Klaus Hildebrandt
TU Delft, NL
- Alec Jacobson
University of Toronto, CA
- Bert Jüttler
Johannes Kepler Universität Linz, AT
- Ladislav Kavan
University of Utah – Salt Lake City, US
- Adarsh Krishnamurthy
Iowa State University, US
- Angela Kunoth
Universität Köln, DE
- David I. W. Levin
University of Toronto, CA
- Angelos Mantzaflaris
INRIA – Valbonne, FR
- Dominik Mokriš
MTU Aero Engines – München, DE
- Jörg Peters
University of Florida – Gainesville, US
- Francesca Pitolli
Sapienza University of Rome, IT
- Ulrich Reif
TU Darmstadt, DE
- Maria Lucia Sampoli
University of Siena, IT
- Thomas Takacs
Johannes Kepler Universität Linz, AT
- Etienne Vouga
University of Texas – Austin, US
- Chris Wojtan
IST Austria – Klosterneuburg, AT
- Urška Zore
MTU Aero Engines – München, DE