

Adjoint Methods in Computational Science, Engineering, and Finance

Edited by

Nicolas R. Gauger¹, Michael Giles², Max Gunzburger³, and Uwe Naumann⁴

1 TU Kaiserslautern, DE, nicolas.gauger@scicomp.uni-kl.de

2 University of Oxford, GB, mike.giles@maths.ox.ac.uk

3 Florida State University, US, mgunzburger@fsu.edu

4 RWTH Aachen University, DE, naumann@stce.rwth-aachen.de

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 14371 “Adjoint Methods in Computational Science, Engineering, and Finance”.

The development of adjoint numerical methods yields a large number of theoretical, algorithmic, and practical (implementation) challenges most of them to be addressed by state of the art Computer Science and Applied Mathematics methodology including parallel high-performance computing, domain-specific program analysis and compiler construction, combinatorial scientific computing, numerical linear algebra / analysis, and functional analysis. One aim of this seminar was to tackle these challenges by setting the stage for accelerated development and deployment of such methods based on in-depth discussions between computer scientists, mathematicians, and practitioners from various (potential) application areas. The number of relevant issues is vast, thus asking for a series of meetings of this type to be initiated by this seminar. It focused on fundamental theoretical issues arising in the context of “continuous vs. discrete adjoints.” The relevant context was provided by presentations of various (potential) applications of adjoint methods in CSEF.

Seminar September 8–12, 2014 – <http://www.dagstuhl.de/14371>

1998 ACM Subject Classification G.1.0 Numerical Algorithms, G.1.4 Automatic Differentiation, G.1.6 Optimization, G.1.7 Ordinary Differential Equations, G.1.8 Partial Differential Equations, G.1.10 Applications, G.4 Mathematical Software

Keywords and phrases continuous adjoints, discrete adjoints, high-performance scientific computing, algorithmic differentiation

Digital Object Identifier 10.4230/DagRep.4.9.1

1 Executive Summary

Uwe Naumann

License  Creative Commons BY 3.0 Unported license
© Uwe Naumann

The human desire for meaningful numerical simulation of physical, chemical, biological, economical, financial (etc.) phenomena in CSEF has been increasing with the growing performance of the continuously improving computer systems. As a result of this development we are (and will always be) faced with a large (and growing) number of highly complex numerical simulation codes that run at the limit of the available HPC resources. These codes



Except where otherwise noted, content of this report is licensed under a Creative Commons BY 3.0 Unported license

Adjoint Methods in Computational Science, Engineering, and Finance, *Dagstuhl Reports*, Vol. 4, Issue 9, pp. 1–29

Editors: Nicolas R. Gauger, Michael Giles, Max Gunzburger, and Uwe Naumann



DAGSTUHL
REPORTS

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

often result from the discretization of systems of PDE. Their run time correlates with the spatial and temporal resolution which often needs to be very high in order to capture the real behavior of the underlying system. There is no doubt that the available hardware will always be used to the extreme. Improvements in the run time of the simulations need to be sought through research in numerical algorithms and their efficient implementation on HPC architectures.

Problem sizes are often in the billions of unknowns; and with emerging large-scale computing systems, this size is expected to increase by a factor of thousand over the next five years. Moreover, simulations are increasingly used in design optimization and parameter identification which is even more complex and requires the highest possible computational performance and fundamental enabling algorithmic technology. Derivatives of certain objectives of these numerical models with respect to a potentially very large number of model parameters are crucial for the highly desirable transition from pure simulation to optimization. Approximation of these derivatives via finite difference quotients often lacks the required accuracy. More importantly, it may be infeasible for a large parameter space in terms of its computational complexity. Adjoint numerical programs have until recently been written by hand to overcome this problem. Such programs compute (large) gradients with machine accuracy at a small constant multiple of the computational complexity of the underlying primal simulation. Due to the enormous size of most numerical simulation codes the manual procedure may take up to several man years. Moreover manual adjoint codes are error-prone and hard to maintain as the primal simulation evolves. Computer scientists have been developing special software tools based on the principles of algorithmic differentiation (AD) to generate discrete adjoint code automatically. Consequently, this method has gained considerable acceptance within the CSEF community as illustrated by numerous successful case studies presented in the proceedings of so far six international conferences on AD. See <http://www.autodiff.org> for details.

Illustrative Example: Classical applications of adjoint methods arise in the context of large-scale inverse problems, such as the estimation of unknown or uncertain parameters of implementations of mathematical models for real-world problems as computer programs. Imagine the optimization of the shape of an aircraft with the objective to maximize its lift. The continuous mathematical domain (the surface of the aircraft) is typically discretized through the generation of a mesh with a potentially very large number of points spread over the whole surface. Optimization aims to adapt the position of these points in 3D space such that the objective is met while at the same time satisfying various constraints (e.g. prescribed volume). A naive approach might run a potentially very large number of primal numerical simulations with changing mesh configurations thus being able to identify an optimum within this very limited search space.

Derivative-based approaches use information on the sensitivity of the objective at the given mesh configuration with respect to changes in the positions of all mesh points (the gradient) in order to make a deterministic decision about the next configuration to be considered. The sensitivities can be approximated through local perturbations of the position of each mesh point (finite difference quotients). A single optimization step would thus require a number of primal simulations that is of the order of the number of degrees of freedom (three spatial coordinates for each mesh point) induced by the mesh. This approach is practically infeasible as a single simulation may easily run for several minutes (if not hours) on the latest HPC architectures. The approximation of a single gradient would take months (if not years) for a mesh with only one million points.

Adjoint methods deliver the gradient at the cost of only a few (between 2 and 10) primal

simulations. Continuous adjoint methods derive an adjoint version of the primal mathematical model analytically followed by the numerical solution of the resulting adjoint model. While this approach promises low computational cost (approx. 2 primal simulations) it can be mathematically challenging and numerically inconsistent when compared with the primal numerical simulation. To the best of our knowledge, the automation of the derivation of continuous adjoint models is still outstanding.

Discrete adjoint methods rely on the algorithmic differentiation of the primal numerical model, thus overcoming the potential numerical inconsistencies induced by the continuous adjoint. Depending on the mode of implementation of AD, the level of maturity of the AD tool, and the expertise of the user of the tool the computational cost can range between 2 and 20 primal simulations, sometimes even more. Still this cost is independent of the number of mesh points (referring to the above example). Solutions to problems arising in adjoint methods require expertise in both theoretical and applied Computer Science as well as in Numerical Analysis. Robust methods for the data flow reversal within adjoint code are built on special graph partitioning and coloring algorithms. Their implementation on modern HPC architectures (e. g. using MPI and/or OpenMP) has impact on the simulation software design and the data management. The use of accelerators has been considered only recently with many open as of yet unsolved problems. Static and dynamic program analysis and compiler construction techniques have been developed to facilitate the semi-automatic generation of discrete adjoint code. The exploration of a potential extension of these techniques to continuous adjoint code was one of the subjects of this seminar. Other conceptual problems discussed included functional analytic aspects of adjoint methods and their impact on practical implementation, combinatorial problems in adjoint code generation and their computational complexities, and simulation software engineering guidelines in the light of adjoint methods.

Adjoint methods borrow from a variety of subfields of Computer Science and Applied Mathematics including high performance and combinatorial scientific computing, program analysis and compiler construction, functional analysis, numerical analysis and linear algebra, and with relevance to a wide range of potential areas of application. As such, the topic lends itself to a series of seminars taking more detailed looks into the respective subjects. With this seminar we intent to initiate a sequence of related events alternating in between the Leibniz Center for Informatics at Schloss Dagstuhl and the Mathematisches Forschungsinstitut Oberwolfach, thus, emphasizing the obvious synergies between Computer Science and Mathematics in the given context.

2 Table of Contents

Executive Summary

<i>Uwe Naumann</i>	1
------------------------------	---

Overview of Talks

OO-Lint for Operator Overloading in C++ <i>Christian Bischof</i>	6
---	---

An extension of the projected gradient method with application in multi-material structural topology optimization <i>Luise Blank</i>	6
---	---

Algorithmic Differentiation for Geometry Processing <i>David Bommers</i>	7
---	---

An interface for conveying high-level user information to automatic differentiation transformations: A working group proposal <i>Martin Bückner</i>	7
--	---

The Glorious Future of Automatic Differentiation: A Retrospective View <i>Bruce Christianson</i>	8
---	---

Adjoint-Based Research and Applications at NASA Langley Research Center <i>Boris Diskin</i>	8
--	---

Automated adjoint finite element simulations within FEniCS <i>Patrick Farrell</i>	9
--	---

Physical Interpretations of Discrete and Continuous Adjoint Boundary Conditions <i>Christian Frey</i>	10
--	----

Efficient Adjoint-based Techniques for Optimal Active Flow Control <i>Nicolas R. Gauger</i>	10
--	----

Integrating and Adjoining ODEs with Lipschitzian RHS <i>Andreas Griewank</i>	11
---	----

Discrete versus Continuous Adjoint of Differential-Algebraic Equation Systems: Similarities and Differences <i>Ralf Hannemann-Tamas</i>	12
--	----

Bridges between OO-based and ST-based adjoint AD <i>Laurent Hascoet</i>	13
--	----

Application of derivative code in climate modeling <i>Patrick Heimbach</i>	14
---	----

Adjoint in Solution Methods for PDE Constrained Inverse Problems: Reduced versus all-at-once formulations <i>Barbara Kaltenbacher</i>	16
--	----

Regularity of Model, Adjoint and its Relation to Optimization (Data Assimilation) <i>Peter Korn</i>	17
--	----

Constraint handling for gradient-based optimization of compositional reservoir flow <i>Drosos Kourounis</i>	18
--	----

Tool-Demo: Algorithmic Differentiation by Overloading in C++ using dco/c++ <i>Johannes Lotz</i>	18
Adjoint Numerical Libraries <i>Viktor Mosenkis</i>	19
Moans about discrete adjoints <i>Jens-Dominik Mueller</i>	19
Adjoint-Based Research and Applications at NASA Langley Research Center <i>Eric Nielsen</i>	20
Goal-oriented mesh adaptation based on total derivative of goal with respect volume mesh coordinates <i>Jacques Peter</i>	21
Numerical Algorithms for Mean-field type control Problems <i>Olivier Pironneau</i>	21
The Edge Pushing Algorithm for Computing Sparse Hessians <i>Alex Pothen</i>	22
Hybrid Adjoint Approaches to Industrial Hydrodynamics <i>Thomas Rung</i>	22
Structure Exploitation, AD and the Continuous Problem <i>Stephan Schmidt</i>	23
Linking Adjoint Based Shape Optimization to Riemannian Geometry in Shape Spaces <i>Volker Schulz</i>	24
Discrete Adjoint Optimization for OpenFOAM <i>Markus Towara</i>	24
Convergence of discrete adjoints for flows with shocks <i>Stefan Ulbrich</i>	25
White box adjoining of a radiative transport model and its use for ill-posed inverse problems <i>Joern Ungermann</i>	26
Time-minimal Checkpointing <i>Andrea Walther</i>	27
Adjoining Chaos <i>Qiqi Wang</i>	28
Participants	29

3 Overview of Talks

3.1 OO-Lint for Operator Overloading in C++

Christian Bischof (TU Darmstadt, DE)

License  Creative Commons BY 3.0 Unported license
© Christian Bischof

Automatic Differentiation Tools in C++ often employ operator overloading (OO), and in particular almost all reverse mode tools employ this technology. This seemingly simple approach just requires a type change in the numeric code, redeclaring the floating point type. However, in practice, it is not so simple, as the operator overloading may lead to conflicts with the C++ standard, such as, for example, multiple user defined conversions, unions with complex types, or implicit conversions in conditions, and, as a result, to cryptic error messages.

To alleviate this problem, we developed a tool based on LLVM/Clang, which recognizes problematic coding constructs in the C++ code to be subjected to operator overloading. This tool thus provides guidance to a potential user of OO-based semantic enhancements of an existing code, in a fashion that is much more targeted and understandable than the usual error messages produced by the compiler. In addition, we are working on automating the changes necessary to make the code applicable to OO, thus further easing the transition for potential users of OO-based AD tools.

3.2 An extension of the projected gradient method with application in multi-material structural topology optimization

Luise Blank (Universität Regensburg, DE)

License  Creative Commons BY 3.0 Unported license
© Luise Blank

Joint work of Blank, Luise; Rupprecht, Christoph

First we introduce the phase field approach for the optimal distribution of several elastic isotropic homogeneous materials. This leads to a multi-material structural topology optimization problem with linear elasticity equations, mass constraints and pointwise inequalities as restrictions. The reduced problem formulation results into a nonlinear optimization problem over a convex and closed set. While the reduced cost functional is Fréchet-differentiable in $H^1 \cap L^\infty$ it is not differentiable in a Hilbert-space. Hence the classical theory for projected gradient methods cannot be applied. Therefore, we extend the projected gradient method to Banach spaces. The gradient is not required but only directional derivatives. Furthermore, variable scaling and varying the metric is introduced. The last allows the use of second order information in the method. We prove global convergence of the method. This method is applied to the presented optimization problem. Here it turns out that the scaling of the derivative with respect to the interface thickness is important to obtain a drastic speed up of the method. With computational experiments we demonstrate the independence of the discretization mesh size and of the interface thickness in the number of iterations as well as its efficiency in time. Moreover we present results for compliance mechanism and drag minimization in Stokes flow.

3.3 Algorithmic Differentiation for Geometry Processing

David Bommes (INRIA Sophia Antipolis – Méditerranée, FR)

License © Creative Commons BY 3.0 Unported license
© David Bommes

Joint work of Bommes, David; Lotz, Johannes; Naumann, Uwe

Recent geometry processing approaches are often related to optimization of complicated nonlinear functionals and constraints. The resulting problems are usually optimized with interior point methods that require second order derivative information. The goal of this project is to develop a framework for rapid prototyping of such applications with the help of algorithmic differentiation, where based on a user-provided functional evaluation, all derivatives should be generated automatically. To obtain a performance that is comparable to the manual approach, it is crucial to exploit the sparsity structure that geometry processing approaches provide due to partial separability [1] of the corresponding functionals. By providing an intuitive interface where the partial separability becomes obvious, we develop a fast and easy to use framework for geometry processing with AD, which integrates C++ overloading techniques like ADOL-C [2] and dco/c++ [3].

References

- 1 Bischof, C. H., Bouaricha, A., Khademi, P. M., Moré, J. J. (1997). Computing gradients in large-scale optimization using automatic differentiation. *INFORMS Journal on Computing*, 9(2), 185–194.
- 2 Walther, A., Griewank, A. (2012). Getting started with ADOL-C. *Combinatorial Scientific Computing*, 181–202.
- 3 Naumann, U. (2011). *The art of differentiating computer programs: an introduction to algorithmic differentiation* (Vol. 24). SIAM.

3.4 An interface for conveying high-level user information to automatic differentiation transformations: A working group proposal

Martin Buecker (Universität Jena, DE)

License © Creative Commons BY 3.0 Unported license
© Martin Buecker

Program transformations that augment a given computer code with statements for the computations of derivatives are commonly referred to as automatic or algorithmic differentiation (AD). Software tools implementing the AD technology are available for various programming languages; see the community web site www.autodiff.org. Today, there are robust AD tools which are capable of correctly transforming large programs with minimal human intervention. However, sometimes, AD is not “automatic.” In fact, a combined approach that applies AD in a black-box fashion to large parts of the code and that also involves a moderate amount of human intervention for certain parts of the code is often adequate. This Dagstuhl seminar is a perfect opportunity to discuss the following questions: Where is human intervention absolutely necessary? What can be handled mechanically by an AD tool? To what extent can the level of abstraction be raised by user-specified directives?

3.5 The Glorious Future of Automatic Differentiation: A Retrospective View

Bruce Christianson (University of Hertfordshire, GB)

License  Creative Commons BY 3.0 Unported license
© Bruce Christianson

The adjoint mode of AD is both fast and accurate. We have had considerable success with selling the advantage of speed to a variety of application communities. To a large extent this is because AD tools have evolved so as to cope well with legacy code. There is still room for improvement, but we should not become fixated on embracing the legacy agenda.

With regard to the second advantage, we have made little headway in selling accuracy as a step-change benefit. This failure is largely due to the continued use of legacy approaches to modeling, and to optimization.

At present, a smooth continuous model is typically discretized, not always consistently, before being implemented (using loops with an epsilon-based stopping criterion) into a program that is not continuous, let alone smooth. Finite difference approximations are used to smooth these discontinuities over the scale of the anticipated step. The resulting values are used by optimization algorithms (such as Quasi-Newton), to build internally a new smooth local model, that is inconsistent with the original model. Finally, this new model is solved exactly.

It is clear that simply differentiating the discontinuous program accurately does not much help overall performance or convergence: users want secants not tangents; the primal problem is seldom converged accurately until near the optimum; the raw models exhibit chaotic behavior; and Newton is not stable.

Louis Rall often pointed out that Automatic Differentiation is not really a local operation. What mileage can we gain by re-factoring AD to obtain accurate, and more importantly, consistent solutions to systematic smooth perturbations of the original model, for example?

As a side-effect, the adjoint mode produces large numbers of potentially useful by-products, such as Lagrange multipliers, for free. However, when solving equations, legacy modeling software often does not explicitly identify, or even calculate, the corresponding equation residuals.

We, the AD community, need urgently to clarify our thinking, and prepare our agenda for changing the next generation of modeling and optimization tools.

3.6 Adjoint-Based Research and Applications at NASA Langley Research Center

Boris Diskin (National Institute of Aerospace – Hampton, US)

License  Creative Commons BY 3.0 Unported license
© Boris Diskin

Joint work of Diskin, Boris; Nielsen, Eric

Main reference E. Nielsen, B. Diskin, “Discrete Adjoint-Based Design for Unsteady Turbulent Flows on Dynamic Overset Unstructured Grids,” AIAA Journal, 51(6):1355–1373, June 2013; available also from NASA repository.

URL <http://dx.doi.org/10.2514/1.J051859>

URL http://fun3d.larc.nasa.gov/papers/overset_adjoint_aiaaj.pdf

An overview of the use of adjoint methods at NASA Langley Research Center has been presented. Among major areas of large-scale adjoint-based applications are shape optimization,

grid adaptation, and multidisciplinary optimization. In this talk, examples of unsteady adjoint-based aerodynamic shape optimization have been presented such as active flow control for a high-lift configuration, a helicopter in forward flight, a fighter jet with simulated aeroelastic effects, and a biologically-inspired flapping wing configuration. Several examples of adjoint-based mesh adaptation for various applications have included high-lift and nozzle plume configurations, a sonic boom application, an example of shock-boundary layer interaction, and several other aerospace applications. Multidisciplinary optimization capabilities have been demonstrated for sonic boom mitigation. High-performance computing aspects have also been discussed, such as scaling performance in the presence of frequent I/O for unsteady adjoint simulations.

3.7 Automated adjoint finite element simulations within FEniCS

Patrick Farrell (University of Oxford, GB)

License © Creative Commons BY 3.0 Unported license
© Patrick Farrell

Joint work of Farrell, Patrick; Funke, Simon; Ham, David; Rognes, Marie; Biriksson, Ásgeir
Main reference P. E. Farrell, D. A. Ham, S. W. Funke, M. E. Rognes, “Automated derivation of the adjoint of high-level transient finite element programs,” *SIAM Journal on Scientific Computing*, 35(4):C369–C393, July 2013.
URL <http://dx.doi.org/10.1137/120873558>

In this work, we develop and advocate a high-level approach to automated adjoint derivation for finite element simulators. By “high-level”, we mean that instead of breaking down a program into a sequence of elementary machine instructions, we treat the program in terms of much higher-level mathematical constructs: in the finite element case, the solution of variational problems. By retaining as much mathematical structure as possible, this approach offers several advantages: the derived tangent linear and adjoint solutions work naturally in parallel; the adjoint solver can automatically use optimal checkpointing schemes with minimal user intervention; and the tangent linear and adjoint models typically exhibit optimal efficiency.

This high-level perspective permits optimizations and modifications that would be impractical in low-level code. For example, when computing Hessian actions, the equations to be solved for each action (the tangent linear and second-order adjoint equations) share the same matrices to be solved, up to transposition: a high-level AD tool can cache the factorizations of these matrices and re-use them to dramatically speed up the per-action cost, but such an optimization would be extremely difficult to implement in a low-level AD tool.

Several examples were presented, including nonlinear diffusion on a manifold; Hessian eigendecomposition for the test case of Deckelnick and Hinze; and mesh independence in solving the mother problem of PDE-constrained optimization.

In general, to attain optimal performance with algorithmic differentiation, the tool needs to exploit all of the mathematical structure available in a given problem domain. This work is specific to finite elements, but it would be entirely possible to apply the experience gained to finite volume discretizations, or problem domains outside of PDEs.

In addition, I presented very recent results on deflation techniques for computing distinct solutions of nonlinear systems and distinct local minima of nonconvex optimization problems.

3.8 Physical Interpretations of Discrete and Continuous Adjoint Boundary Conditions

Christian Frey (DLR – Köln, DE)

License © Creative Commons BY 3.0 Unported license
© Christian Frey

Joint work of Frey, Christian; Engels-Putzka, Kügeler, Edmund
Main reference C. Frey, A. Engels-Putzka, E. Kügeler, “Adjoint boundary conditions for turbomachinery flows,” in Proc. of the 2012 Europ. Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS’12), pp. 9360–9379, 2012.
URL <http://elib.dlr.de/81285/>

In this talk I have discussed the treatment of boundary conditions in the context of a discrete adjoint industrial turbomachinery RANS solver. Special emphasis is put on the non-reflecting boundary conditions and the blade row coupling by mixing planes. These techniques are widely used for the accurate approximation of time-averaged flows in turbomachinery by steady simulations. In contrast to inviscid wall boundary conditions these boundary conditions are applied at non-characteristic boundaries, i. e., the flux Jacobian through the boundary is non-singular, unless the normal flow vanishes at some point of the blade row entry or exit. On the other hand, these boundary conditions are more complicated in that they are non-local. They involve, for instance, Fourier transformations, and special averaging techniques.

We outline a general methodology to adjoin discretely numerical boundary conditions and apply the techniques to the boundary conditions of an internal flow solver. This leads to adjoint boundary update operators which are applied after each multiplication with the adjoint residual Jacobian. This methodology carries over to the communication using domain decomposition and ghost cells.

A further difficulty that one has to deal with is the fact that the boundary conditions in the non-linear solver are implemented as fixed-point iterations, whereas the adjoint system is solved by the GMRES method. This means that the application of AD seems to be rather difficult and may require that one switches to a discrete adjoint iterative solver.

The second part of this talk is dedicated to the physical interpretation of the discrete adjoint boundary update as discretizations of their continuous adjoint counterpart. For this purpose we determine the continuous adjoint boundary conditions for the above-mentioned turbomachinery boundary conditions. Finally we give a physical interpretation of the adjoint non-reflecting condition and the adjoint mixing plane coupling condition. The former can be viewed as a non-reflecting condition for the adjoint Euler equations. The adjoint blade row coupling condition is satisfied if certain circumferential averages of the adjoint fields on both sides of the interfaces agree up to a factor given by the blade count ratio.

3.9 Efficient Adjoint-based Techniques for Optimal Active Flow Control

Nicolas R. Gauger (TU Kaiserslautern, DE)

License © Creative Commons BY 3.0 Unported license
© Nicolas R. Gauger

Joint work of Gauger, Nicolas R.; Wang, Qiqi; Kramer, Felix; Thiele, Frank; Günther, Stefanie; Sagebaum, Max

For efficient optimal active control of unsteady flows, the use of adjoint approaches is a first essential ingredient. We compare continuous and discrete adjoint approaches in terms of accuracy, efficiency and robustness. For the generation of discrete adjoint solvers, we discuss

the use of Automatic Differentiation (AD) and its combination with checkpointing techniques. Furthermore, we discuss so-called one-shot methods. Here, one achieves simultaneously convergence of the primal state equation, the adjoint state equation as well as the design equation. The direction and size of the one-shot optimization steps are determined by a carefully selected design space preconditioner. The one-shot method has proven to be very efficient in optimization with steady partial differential equations (PDEs). Applications of the one-shot method in the field of aerodynamic shape optimization with steady Navier-Stokes equations have shown, that the computational cost for an optimization, measured in runtime as well as iteration counts, is only 2 to 8 times the cost of a single simulation of the governing PDE. We present a framework for applying the one-shot approach also to optimal control problems with unsteady Navier-Stokes equations. Straight forward applications of the one-shot method to unsteady problems have shown, that its efficiency depends on the resolution of the physical time domain. In order to dissolve this dependency, we consider unsteady model problems and investigate an adaptive time scaling approach.

3.10 Integrating and Adjoining ODEs with Lipschitzian RHS

Andreas Griewank (HU Berlin, DE)

License © Creative Commons BY 3.0 Unported license
© Andreas Griewank

Joint work of Griewank, Andreas; Lenser, Ben; Hasenfelder, Richard; Streubel, Tom; Boeck, Paul; Gompil, Battur
Main reference P. Boeck, B. Gompil, A. Griewank, R. Hasenfelder, N. Strogies, “Experiments with generalized midpoint and trapezoidal rules on two nonsmooth ODE’s,” *Mong. Math. J.*, 17:39–49, 2013.

URL <http://iom.num.edu.mn/journal/2013/3.pdf>

We consider initial value problems in ODEs where the right hand side has truly state-dependent kinks or jumps and users may be hard pressed to provide suitable switching functions for the customary event handling approaches. Instead kinks and jumps can be detected and handled automatically, which is possible by an extension of algorithmic differentiation that provides piecewise linear approximations with second order error to piecewise smooth and Lipschitz continuous right hand sides [3]. Without continuity of the underlying piecewise smooth function the resulting piecewise linear approximation will also be discontinuous and the approximation error is no longer uniform but heavily direction dependent. Nevertheless we expect to extend our approach later to ODEs where the RHS has jumps but the exact solution trajectories satisfy a certain transversality condition. In the Lipschitzian case we show how the piecewise linearizations of the RHS is generated by Algorithmic Piecewise Differentiation abs-normal form and how it can be used to generalize the midpoint and the trapezoidal rule such that local third order consistency and uniform global convergence order two is recovered [1]. The inherent smoothness of the approximation facilitates the gain of one or two extra orders by Richardson/Romberg extrapolation. The two implicit discretizations of the ODEs produce non-smooth systems of algebraic equations which have been solved by the methods discussed in [4]. The corresponding adjoint trajectory is defined by a differential inclusion and thus not unique if there are valley tracing modes as defined by P. Barton and K. Khan [2]. Nevertheless, usually one obtains a generalized gradient that can be used for data assimilation and more general optimal control. We report preliminary results on a shallow water equation in 1D where non smoothness arises through slope limiters or no smooth norms in the error functional.

References

- 1 P. Boeck, B. Gompil, A. Griewank, R. Hasenfelder, and N. Strogies *Experiments with generalized midpoint and trapezoidal rules on two nonsmooth ODE's*, Mong. Math. J., Vol. 17, pp. 39–49. (2013)
- 2 Kamil A. Khan, Paul I. Barton *Switching behavior of solutions of ordinary differential equations with nonsmooth right-hand sides* (2014)
- 3 Andreas Griewank. *On Stable Piecewise Linearization and Generalized Differentiation*. Optimization Methods and Software, 28(6):1139–1178 (2013)
- 4 A. Griewank, J.-U. Bernt, M. Radons, and T. Streubel. *Solving piecewise linear equations in abs-normal form*. Optimization-Online 2013/12/4184 (2013).

3.11 Discrete versus Continuous Adjoints of Differential-Algebraic Equation Systems: Similarities and Differences

Ralf Hannemann-Tamas (Univ. of Science & Technology – Trondheim, NO)

License © Creative Commons BY 3.0 Unported license
© Ralf Hannemann-Tamas

Main reference Ralf Hannemann-Tamas, “Adjoint Sensitivity Analysis for Optimal Control of Non-Smooth Differential-Algebraic Equations,” Dissertation, Shaker Verlag, 2013.

URL <http://www.shaker.de/de/content/catalogue/index.asp?lang=de&ISBN=978-3-8440-1635-2>

This work is based on the PhD thesis [1]. In contrast to multi-step methods [2], one step-methods applied to semi-explicit DAE systems of index 1, result in differential discrete adjoints which are consistent with the differential continuous adjoints, while the discrete adjoints for algebraic variables tend to zero along their trajectory.

We illustrate this fact by a small example. Let R denote the field of the real numbers and let $\psi : R^{n_x} \rightarrow R$ be a smooth function. We aim to compute the gradient of the scalar functional J , where $J(x, y, p) := \psi(x(t_1))$, with respect to the parameter vector p , where $x(t_1)$ is characterized by the parametric initial value problem

$$\begin{aligned} \dot{x} &= f(x, y), \\ 0 &= g(x, y), \\ x(t_0) &= p, \\ y(t_0) &= y_0. \end{aligned}$$

Here, $x(t) \in R^{n_x}$, $y(t) \in R^{n_y}$, $p \in R^{n_p}$ denote the differential variables, algebraic variables and the parameters, respectively. The initial and final times are t_0 and t_1 , respectively and the mappings $f : R^{n_x} \times R^{n_y} \rightarrow R^{n_x}$ and $g : R^{n_x} \times R^{n_y} \rightarrow R^{n_y}$ are assumed to be sufficiently smooth. Further, the initial value problem is assumed to have a unique solution.

For the moment we assume that initial values y_0 are consistent, i. e., they are a solution of the algebraic equation

$$g(p, y_0) = 0.$$

Let x_1, y_1 approximate the differential and algebraic variables at time t_1 as the solution of the one-step method (in Henrici's notation)

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} p \\ y_0 \end{pmatrix} + \Phi(t_0, x_0, y_0, h), \quad \text{with } h = t_1 - t_0.$$

Then, the discrete adjoints λ_0^x, λ_0^y at time t_0 can be interpreted as derivatives of the objective with respect to the initial values

$$\lambda_0^x = \frac{\partial\psi(x_1)}{\partial p}, \quad \lambda_0^y = \frac{\partial\psi(x_1)}{\partial y_0}. \quad (1)$$

Especially, the adjoints λ_0^y , associated with the algebraic variables, describe the sensitivity of the objective function with respect to perturbations δy_0 of the then inconsistent initial values $\lambda_0^y = y_0 + \delta y_0$. However, a good numerical method evens out small inconsistencies of algebraic initial values. Hence, the values of the algebraic adjoints λ_0^y tend to zero, i. e. $\lambda_0^y \approx 0$. In particular, at $t = t_0$, the discrete adjoints λ_0^y deviate from continuous algebraic adjoints $\lambda^y(t_0)$, since they are usually different from zero (e.g. see [3]). In contrast, the differential discrete adjoints λ_0^x usually converge with high order against the differential continuous adjoints $\lambda^x(t_0)$, since these satisfy (e.g. see [3])

$$\lambda^x(t_0) = \frac{\partial\psi(x(t_1))}{\partial p},$$

which is analogous to the first identity in eq. (1).

To summarize, the similarity is that differential discrete adjoints converge (for $h \rightarrow 0$) to their continuous counterparts. The difference is, that the latter convergence result does not hold for algebraic discrete adjoints.

References

- 1 R. Hannemann-Tamás. *Adjoint Sensitivity Analysis for Optimal Control of Non-Smooth Differential-Algebraic Equations*. Dissertation, Shaker Verlag, 2013.
- 2 A. Sandu. *Reverse automatic differentiation of linear multistep methods*. In T. J. Barth, M. Griebel, D. E. Keyes, R. M. Nieminen, D. Roose, T. Schlick, C. H. Bischof, H. M. Bücker, P. Hovland, U. Naumann, and J. Utke (Eds.), *Advances in Automatic Differentiation*, Volume 64 of Lecture Notes in Computational Science and Engineering, pp. 1–12. Springer, 2008.
- 3 Y. Cao, S. Li, L. Petzold, and R. Serban. *Adjoint sensitivity analysis for differential-algebraic equations: The adjoint DAE system and its numerical solution*. *SIAM J. Sci. Comput.* 24(3), pp. 1076–1089, 2003.

3.12 Bridges between OO-based and ST-based adjoint AD

Laurent Hascoet (INRIA Sophia Antipolis – Méditerranée, FR)

License © Creative Commons BY 3.0 Unported license
© Laurent Hascoet

Among AD tools, there is a clear opposition between those based on Operator Overloading (OO) and those based on Source Transformation (ST). Competition between the two classes of tools has brought huge improvements to both but obviously no class will ever show definitive superiority. We claim that this competition, fruitful as it was, must give way to collaboration between OO-based and ST-based AD models.

We believe that the difference between OO-based and ST-based adjoint AD is not as deep as one may think at first sight. Indeed we think that the vocabulary and techniques of Partial Evaluation can help us exhibit a close resemblance, not only at some conceptual level, but also leading to fruitful exchange of techniques between OO-based and ST-based.

In the framework of Partial Evaluation, we can view the tape, built by the tape-recording phase of OO-based adjoint differentiation, as made of two parts, one part being static i. e. depending only on the program to differentiate, the other being dynamic i. e. depending also on the particular input. Obviously the static part could be extracted once and for all from the program to differentiate. By Partial Evaluation of the Tape Interpreter with respect to the static part of the tape, one obtains a Specialized Tape Interpreter that, when given the dynamic part, will evaluate the adjoint derivatives more efficiently than the initial Tape Interpreter. Also, the dynamic part of the Tape is smaller than the full tape, and can actually be much smaller. We claim that the computation of the dynamic part of the tape on one hand and the execution of the Specialized Tape Interpreter on the other hand, correspond exactly to the two phases of a ST-based adjoint namely, the forward sweep on one hand and the backward sweep on the other hand.

Practically, we think we tool developers should explore more ways to make OO-based and ST-based AD tools collaborate. One key architectural choice being “who is in the driver’s seat”. For instance black-box mechanisms in OO-based environments allow them to call ST-based adjoints for computational kernels where the language constructs pose no difficulty to static source analysis and transformation. Only this is still a tedious process. Conversely, there must be ways for OO-based AD to take better advantage of the static data-flow analysis (activity, liveness, TBR, . . .) that ST-based AD computes and uses routinely. In particular the activity analysis of an ST-based tool can be used to automate the type transformation stage that an OO-based AD used must perform by hand.

3.13 Application of derivative code in climate modeling

Patrick Heimbach (MIT – Cambridge, US)

License © Creative Commons BY 3.0 Unported license
© Patrick Heimbach

Main reference C. Wunsch, P. Heimbach, “Dynamically and kinematically consistent global ocean circulation and ice state estimates,” chapter 21 in G. Siedler, J. Church, J. Gould, S. Griffies, eds., “Ocean Circulation and Climate: A 21st Century Perspective,” pp. 553–579, Elsevier, 2013.

Main reference <http://store.elsevier.com/product.jsp?isbn=9780123918536>
URL <http://ecco-group.org>

Optimal state and parameter estimation, accompanied by rigorous uncertainty quantification, is increasingly being recognized as a powerful tool in climate modeling. The need arises in order to deal with the problem of sparse observations to optimally constrain model simulations and infer dynamically consistent time-evolving state estimates (Heimbach and Wunsch 2012), to provide quantitative estimates of the extent to which existing observations constrain uncertain parameters or the optimal design of future observing networks (Heimbach et al. 2010), or to infer optimal initial conditions that are best suited for predictions or projections (Zanna et al. 2012). The primary example given is that of estimating the global ocean (and sea ice) circulation over the last few decades as undertaken by the “Estimating the Circulation and Climate of the Ocean” (ecco-group.org) consortium (Stammer et al. 2002; Wunsch and Heimbach 2007, 2013, 2014).

A number of regional efforts targeting higher spatial resolutions and shorter time scales are also being pursued to synthesize the available data with the known dynamics, e. g., in the North Atlantic (Ayoub 2006; Gebbie et al. 2006), the Southern Ocean (Mazloff et al. 2010), the tropical Pacific (Hoteit et al. 2010), or the Labrador Sea and Baffin Bay (Fenty and Heimbach 2013).

Similar efforts by a number of groups are now targeting the polar ice sheets for the purpose of developing predictive capabilities and uncertainty estimates of ice sheet mass loss in the coming centuries (Heimbach and Bugnion 2009; Goldberg and Heimbach 2013; Larour et al. 2014; Perego et al. 2014).

Increasingly, second derivative, i. e. Hessian information is being explored to provide a posteriori uncertainty estimates on addition to the maximum a posteriori probability estimate, and to propagate these uncertainties forward onto target quantities of interest, e. g. climate indices (Kalmikov and Heimbach 2014; Petra et al. 2014).

In order to improve accessibility to algorithmic differentiation (AD) tools that support flexible derivative code development, the ECCO group has been involved in the development, at Argonne National Lab, of the open source AD tool OpenAD (Naumann et al. 2006; Utke et al. 2008, 2009). Today, a number of configurations of the MIT general circulation model (mitgcm.org) are available for AD-enhanced simulations.

References

- 1 Ayoub, N. (2006). Estimation of boundary values in a North Atlantic circulation model using an adjoint method. *Ocean Modeling*, 12(3-4), 319–347. doi:10.1016/j.ocemod.2005.06.003
- 2 Fenty, I.G. and P. Heimbach (2013). Coupled Sea Ice-Ocean State Estimation in the Labrador Sea and Baffin Bay. *J. Phys. Oceanogr.*, 43(6), 884–904, doi:10.1175/JPO-D-12-065.1.
- 3 Gebbie, G., P. Heimbach and C. Wunsch, 2006: Strategies for nested and eddy-resolving state estimation. *J. Geophys. Res.*, 111, C10073, doi:10.1029/2005JC003094.
- 4 Goldberg, D.N. and P. Heimbach (2013). Parameter and state estimation with a time-dependent adjoint marine ice sheet model. *The Cryosphere*, 7, 1659–1678, doi:10.5194/tc-7-1659-2013.
- 5 Heimbach, P. and C. Wunsch (2012). Decadal ocean (and ice) state estimation for climate research: What are the needs? *Oberwolfach Reports*, 9(4), 3451–3454, doi:10.4171/OWR/2012/58
- 6 Heimbach, P. and V. Bugnion (2009). Greenland ice sheet volume sensitivity to basal, surface, and initial conditions, derived from an adjoint model. *Annals of Glaciology*, 50(52), 67–80, doi:10.3189/172756409789624256
- 7 Heimbach, P., G. Forget, R. Ponte, and C. Wunsch, et al. (2010). Observational Requirements for global-scale ocean climate analysis: Lessons from ocean state estimation. *Community White Paper*. In: Hall, J., D.E. Harrison, and D. Stammer (Eds.), 2010: Proc. of OceanObs 09: Sustained Ocean Observations and Information for Society, Venice, Italy, 21–25 September 2009, ESA Publication WPP-306, Vol. 2, doi:10.5270/OceanObs09.cwp.42.
- 8 Heimbach, P., C. Hill and R. Giering (2002). Automatic Generation of Efficient Adjoint Code for a Parallel Navier-Stokes Solver. in: J.J. Dongarra, P.M.A. Sloot and C.J.K. Tan (Eds.), *Lecture Notes in Computer Science (LNCS)*, Vol. 2330, part II, pp. 1019–1028, Springer-Verlag, doi:10.1007/3-540-46080-2_107.
- 9 Hoteit, I., B. Cornuelle, and P. Heimbach, 2010: An Eddy-Permitting, Dynamically Consistent Adjoint-Based Assimilation System for the Tropical Pacific: Hindcast Experiments in 2000. *J. Geophys. Res.*, 115, C03001, doi:10.1029/2009JC005437.
- 10 Kalmikov, A. and P. Heimbach (2014). A Hessian-based method for Uncertainty Quantification in Global Ocean State Estimation. *SIAM J. Scientific Computing (Special Section on Planet Earth and Big Data)*, 36(5), S267–S295, doi:10.1137/130925311.
- 11 Larour, E., Utke, J., Csatho, B., Schenk, A., Seroussi, H., Morlighem, M., Rignot, E., Schlegel, N., and Khazendar, A. (2014). Inferred basal friction and surface mass balance of North-East Greenland Ice Stream using data assimilation of ICESat-1 surface altimetry and ISSM, *The Cryosphere Discuss.*, 8, 2331–2373, doi:10.5194/tcd-8-2331-2014.

- 12 Mazloff, M.R., P. Heimbach and C. Wunsch (2010). An Eddy-Permitting Southern Ocean State Estimate. *J. Phys. Oceanogr.*, 40(5), 880–899, doi:10.1175/2009JPO4236.1.
- 13 Naumann, U., J. Utke, P. Heimbach, C. Hill, D. Ozyurt, C. Wunsch, M. Fagan, N. Thallent, M. Strout (2006). Adjoint code by source transformation with OpenAD/F. In: *Proc. of the Europ. Conf. on Computational Fluid Dynamics (ECCOMAS CFD 2006)*. P. Wesseling, J. Periaux, and E. Onate (Eds.), TU Delft, The Netherlands, 2006.
- 14 Perego, M., Price, S., and Stadler, G. (2014). Optimal initial conditions for coupling ice sheet models to Earth system models. *Journal of Geophysical Research: Earth Surface*, 119(9), 1894–1917. doi:10.1002/2014JF003181
- 15 Petra, N., Martin, J., Stadler, G., and Ghattas, O. (2014). A Computational Framework for Infinite-Dimensional Bayesian Inverse Problems, Part II: Stochastic Newton MCMC with Application to Ice Sheet Flow Inverse Problems. *SIAM Journal on Scientific Computing*, 36(4), A1525–A1555. doi:10.1137/130934805
- 16 Stammer, D., C. Wunsch, R. Giering, C. Eckert, P. Heimbach, J. Marotzke, A. Adcroft, C. Hill, J. and J. Marshall (2002). The global ocean circulation during 1992–1997, estimated from ocean observations and a general circulation model. *J. Geophys. Res.*, 107 (C9), 3118, doi:10.1029/2001JC000888
- 17 Utke, J., L. Harscoet, P. Heimbach, C. Hill, P. Hovland and U. Naumann (2009). Toward adjointable MPI. *Proceedings of the 10th IEEE International Workshop on Parallel and Distributed Scientific and Engineering, PDSEC-09*, Rome, Italy, pp. 1–8, doi:10.1109/IPDPS.2009.5161165.
- 18 Utke, J., U. Naumann, M. Fagan, N. Thallent, M. Strout, P. Heimbach, C. Hill and C. Wunsch (2008). OpenAD/F: A modular, open-source tool for automatic differentiation of Fortran codes. *ACM Transactions on Mathematical Software (TOMS)*, 34(4), doi:10.1145/1377596.1377598
- 19 Wunsch, C. and P. Heimbach (2014). Bidecadal Thermal Changes in the Abyssal Ocean. *J. Phys. Oceanogr.*, 44(8), 2013–2030, doi:10.1175/JPO-D-13-096.1.
- 20 Wunsch, C. and P. Heimbach (2013). Dynamically and kinematically consistent global ocean circulation and ice state estimates. In: G. Siedler, J. Church, J. Gould and S. Griffies, eds.: *Ocean Circulation and Climate: A 21st Century Perspective*. Chapter 21, pp. 553–579, Elsevier, doi:10.1016/B978-0-12-391851-2.00021-0.
- 21 Wunsch, C. and P. Heimbach (2007). Practical global ocean state estimation. *Physica D*, 230(1-2), pp. 197–208, doi:10.1016/j.physd.2006.09.040.
- 22 Zanna L., P. Heimbach, A. M. Moore and E. Tziperman (2012). Upper-ocean singular vectors of the North Atlantic climate with implications for linear predictability and variability. *Quart. J. Roy. Met. Soc.*, 138(663), 500–513, doi:10.1002/qj.937.

3.14 Adjoints in Solution Methods for PDE Constrained Inverse Problems: Reduced versus all-at-once formulations

Barbara Kaltenbacher (Universität Klagenfurt, AT)

License © Creative Commons BY 3.0 Unported license
© Barbara Kaltenbacher

Main reference B. Kaltenbacher, A. Kirchner, B. Vexler, “Goal oriented adaptivity in the IRGNM for parameter identification in PDEs II: all-at-once formulations,” *Inverse Problems*, 30(4):045002, 2014

URL <http://dx.doi.org/10.1088/0266-5611/30/4/045002>

Inverse problems for partial differential equation such as identification of coefficients, boundary conditions or source terms appear in a wide range of applications. Due to their inherent instability, regularization has to be applied. When computing regularized solutions, adjoints

naturally appear: In minimization based regularization methods like Tikhonov within the first order optimality conditions; for iterative (Newton type or gradient) regularization methods directly in the definition of each step.

The system from which parameters are to be identified typically consist of two parts: The model equation, e.g. a (system of) ordinary or partial differential equation(s), and the observation equation. In the conventional reduced setting, the model equation is eliminated via the parameter-to-state map. Alternatively, one might consider both sets of equations (model and observations) as one large system, to which some regularization method is applied. The choice of the formulation (reduced or all-at-once) can make a large difference computationally, depending on which regularization method is used: Whereas almost the same optimality system arises for the reduced and the all-at-once Tikhonov method, the situation is different for Landweber (LW), i. e., gradient methods: A reduced LW iteration requires solution of the PDE and the (linear) adjoint in each step, whereas in all-at-once LW, only PDE residuals have to be evaluated, but no PDEs need to be solved. In between lie Newton type methods, whose reduced versions again need PDE and adjoint PDE solutions in each step, whereas all-at once versions work with linear PDE (and adjoint) solves only. For the latter we refer to [1].

References

- 1 B. Kaltenbacher, A. Kirchner, B. Vexler: Goal oriented adaptivity in the IRGNM for parameter identification in PDEs II: all-at-once formulations. *Inverse Problems* 30 (2014) 045002.

3.15 Regularity of Model, Adjoint and its Relation to Optimization (Data Assimilation)

Peter Korn (MPI für Meteorologie – Hamburg, DE)

License  Creative Commons BY 3.0 Unported license
© Peter Korn

The problem of determining for a coupled set of nonlinear partial differential equations the initial condition from which a model trajectory emerges in agreement with a given set of time-distributed observations is studied by using a variational data assimilation approach. The partial differential equations describe a simplified coupled Atmospheric-Ocean model and consist of a coupled set of shallow-water equation in geophysical appropriate scaling. For the coupled model the existence of optimal initial conditions in the sense of minimizers of a specific cost functional and a first-order necessary condition involving the coupled adjoint equations are proven. Instrumental for the results are derivative based norms in the data assimilation cost functional such that Sobolev norms replace the standard Lebesgue norms.

References

- 1 Peter Korn. Fitting A Coupled Atmosphere-Ocean Model to Observations using Derivative-Based Norms, preprint, 2014

3.16 Constraint handling for gradient-based optimization of compositional reservoir flow

Drosos Kourounis (University of Lugano, CH)

License © Creative Commons BY 3.0 Unported license
© Drosos Kourounis

Joint work of Kourounis, Drosos; Durlofsky, Louis; Jansen, Jan Dirk; Aziz, Khalid
Main reference D. Kourounis, L. J. Durlofsky, J. D. Jansen, K. Aziz, “Adjoint formulation and constraint handling for gradient-based optimization of compositional reservoir flow,” *Journal of Computational Geosciences*, 18(2):117–137, 2014.
URL <http://dx.doi.org/10.1007/s10596-013-9385-8>

The development of adjoint gradient-based optimization techniques for general compositional flow problems is much more challenging than for oil-water problems due to the increased complexity of the code and the underlying physics. An additional challenge is the treatment of non smooth constraints, an example of which is a maximum gas rate specification in injection or production wells, when the control variables are well bottom-hole pressures. Constraint handling through lumping is a popular and efficient approach. It introduces a smooth function that approximates the maximum of the specified constraints over the entire model or on a well-by-well basis. However, it inevitably restricts the possible solution paths the optimizer may follow preventing it to converge to feasible solutions exhibiting higher optimal values. A simpler way to force feasibility, when the constraints are upper and lower bounds on output quantities, is to satisfy these constraints in the forward model. This heuristic treatment has been demonstrated to be more efficient than lumping and at the same time it obtained better feasible optimal solutions for several models of increased complexity. In this work a new formal constraint handling approach is presented. Necessary modifications of the nonlinear solver used at every time step during the forward simulation are also discussed. All these constrained handling approaches are applied in a gradient-based optimization framework for exploring optimal CO₂ injection strategies that enhance oil recovery for a realistic offshore field, the Norne field. This alternative approach increases the oil production twofold over even the best of its respective competitors.

3.17 Tool-Demo: Algorithmic Differentiation by Overloading in C++ using dco/c++

Johannes Lotz (RWTH Aachen University, DE)

License © Creative Commons BY 3.0 Unported license
© Johannes Lotz

Joint work of Lotz, Johannes; Leppkes, Klaus; Naumann, Uwe
URL https://www.stce.rwth-aachen.de/software/dco_cpp.html

Algorithmic Differentiation (AD) is a widespread technique for the automatic generation of discrete adjoint codes.

AD can be applied by a compiler as a source-to-source transformation or by making use of operator overloading techniques as a built-in language feature. The AD community agrees on the fact that both ways come with advantages and disadvantages. The main advantage of operator overloading is the out-of-the-box coverage of the complete programming language. The efficiency of the generated code on the other hand is its disadvantage, and simultaneously the advantage of a source-to-source compiler. The coupling of both techniques is apparent and required. Nevertheless, no automatic technical solution is available.

In contrast to that, it has not yet become apparent, if the discrete or continuous adjoint approach is preferable. The main difference between the two ways of deriving the adjoint

(the dual) is that the discrete approach inherits all discretization methods from the original problem (the primal). This results in a dual implementation, which is equivalent to a line-by-line derivative of the primal implementation and can therefore be generated by AD techniques. The continuous adjoint approach on the other hand assumes validity of the mathematical equations and on that basis derives the sensitivity equations, which are then to be solved. Discretization decision are to be made again.

Taking both observations into account, a valuable overloading tool should have a flexible and extensible interface to couple not only compiler generated code with the overloading tool, but also hand-written code, eventually being continuous adjoint solutions.

dco/c++ features multiple ways of teaching the tool adjoint knowledge on different levels of abstraction to support the user in implementing the different couplings described above.

3.18 Adjoint Numerical Libraries

Viktor Mosenkis (RWTH Aachen University, DE)

License © Creative Commons BY 3.0 Unported license
© Viktor Mosenkis

Numerical libraries are often used by scientist while writing their codes. This allows them to write the code faster and concentrate on their research rather than spend time on implementing and testing numerical algorithms. Once it comes to adjoint the code the users of these libraries have to solve the problem of providing adjoint version of the numerical library routines. Using Algorithmic Differentiation (AD) may fail because the source code of the routine is not available while writing a continuous adjoint version of the routine as described in [1] is not an easy assignment and requires a deeper insight into the algorithms. And there is still the problem of testing the code. In any case the library supplier is the natural instance for providing adjoint version of his library routines.

The Numerical Algorithm Group delivers adjoint version of their Fortran and C Library routines. AD tool dco/c++ is used to adjoint these routines. Two interfaces are offered. One interface for direct and fast integration in dco/c++. The other one to be used without dco/c++

References

- 1 Uwe Naumann and Johannes Lotz and Klaus Leppkes and Markus Towara. *Algorithmic Differentiation of Numerical Methods: Tangent-Linear and Adjoint Solvers for Systems of Nonlinear Equations*. AIB-2012-15, Aachen, Germany, 2012

3.19 Moans about discrete adjoints

Jens-Dominik Mueller (Queen Mary University of London, GB)

License © Creative Commons BY 3.0 Unported license
© Jens-Dominik Mueller

Joint work of Mueller, Jens-Dominik; Shenren Xu; Marcus Meyer

Main reference S. Xu, D. Radford, M. Meyer, J.-D. Müller, “Stabilization of discrete steady adjoint solvers,” submitted to JCP.

Discrete adjoints can very be produced very effectively with AD tools and promise to provide exact derivatives of the primal code. The former is essential for code maintenance and

evolution: e.g. continuous adjoints typically are much less developed than their primal counterparts. The latter is relevant for advanced adjoint applications such as co-Kriging or uncertainty analysis. However there are pitfalls for discrete adjoints, two of which are highlighted.

For industrial CFD applications where the primal already is close to exhausting the available hardware, the preferred approach is the steady-state adjoints of the steady-state primals of industrial CFD applications. Quite frequently though an adjoint based on a fixed-point paradigm diverges since the primal converges only to limit cycle oscillations, as it is not contractive but its Jacobian possesses unstable eigenvalues. Using a stronger preconditioner can achieve convergence for primal and adjoint in cases of mild instability if used for both primal and adjoint.

On the other hand, counterexamples with continuous adjoints demonstrate that the added stabilization can help to converge the steady-state adjoint even on time-averaged unsteady primals. While a full unsteady approach with checkpointing of the primal will succeed for moderately chaotic flows, adding stabilization to discrete adjoints and quantifying this error should be considered as a robust and very cost-effective way to simulate highly turbulent flows, which in turn also may avoid issues of blow-up related to chaotic behavior.

Issues also arise with the use of point values of the discrete adjoint. Consistent continuous adjoint formulations converge to the analytic adjoint solution, simple examples of uniform channel flows with either Dirichlet or Neumann conditions demonstrate that the discrete adjoint based on a standard primal discretization does not. This in turn precludes the use of point values (rather than integrals) of the adjoint solution, or e.g. the use of boundary formulations of the sensitivity.

While finite-element discretizations naturally possess some dual consistency, for finite volume methods the implications of dual inconsistency and the possible gains offered by modified primals that provide dual-consistent discrete adjoints need more exploration.

3.20 Adjoint-Based Research and Applications at NASA Langley Research Center

Eric Nielsen (NASA Langley ASDC – Hampton, US)

License  Creative Commons BY 3.0 Unported license
© Eric Nielsen

Joint work of Nielsen, Eric; Diskin, Boris

Main reference E. Nielsen, B. Diskin, “Discrete Adjoint-Based Design for Unsteady Turbulent Flows on Dynamic Overset Unstructured Grids,” *AIAA Journal*, 51(6):1355–1373, June 2013; available also from NASA repository.

URL <http://dx.doi.org/10.2514/1.J051859>

URL http://fun3d.larc.nasa.gov/papers/overset_adjoint_aiaaj.pdf

An overview of the use of adjoint methods at NASA Langley Research Center has been presented. Among major areas of large-scale adjoint-based applications are shape optimization, grid adaptation, and multidisciplinary optimization. In this talk, examples of unsteady adjoint-based aerodynamic shape optimization have been presented such as active flow control for a high-lift configuration, a helicopter in forward flight, a fighter jet with simulated aeroelastic effects, and a biologically-inspired flapping wing configuration. Several examples of adjoint-based mesh adaptation for various applications have included high-lift and nozzle plume configurations, a sonic boom application, an example of shock-boundary layer interaction, and several other aerospace applications. Multidisciplinary optimization capabilities have

been demonstrated for sonic boom mitigation. High-performance computing aspects have also been discussed, such as scaling performance in the presence of frequent I/O for unsteady adjoint simulations.

3.21 Goal-oriented mesh adaptation based on total derivative of goal with respect volume mesh coordinates

Jacques Peter (ONERA – Châtillon, FR)

License © Creative Commons BY 3.0 Unported license
© Jacques Peter

Main reference M. Nguyen-Dinh, J. Peter, R. Sauvage, M. Meaux, J.-A. Désidéri, “Mesh quality assessment based on aerodynamic functional output total derivatives,” *European Journal of Mechanics – B/Fluids*, 45(May–June 2014):51–71, 2014.

URL <http://dx.doi.org/10.1016/j.euromechflu.2013.12.004>

In aeronautical CFD, engineers require accurate predictions of the forces and moments but they are less concerned with flow-field accuracy. Hence, the so-called “goal oriented” mesh adaptation strategies have been introduced to get satisfactory values of functional outputs at an acceptable cost, using local node displacement and insertion of new points rather than mesh refinement guided by uniform accuracy. Most often, such methods involve the adjoint vector of the function of interest. Our purpose is to present a new goal oriented criterion of mesh quality and a new local mesh adaptation strategy in the framework of finite-volume schemes and a discrete adjoint vector method. They are based on the total derivative of the goal with respect to mesh nodes coordinates. More precisely, a projection of the goal derivative, removing all components corresponding to geometrical changes in the solid walls or the support of the output is introduced. The norm of this vector field times the local characteristic mesh size is the proposed mesh adaptation criterion. The method is assessed in the case of 2D and 3D Euler flow computations.

3.22 Numerical Algorithms for Mean-field type control Problems

Olivier Pironneau (UPMC – Paris, FR)

License © Creative Commons BY 3.0 Unported license
© Olivier Pironneau

Main reference M. Lauriere, O. Pironneau, “Dynamic programming for mean-field type control,” *C.R.A.S Serie I*, 1–7, Oct. 2014.

URL <http://hal.archives-ouvertes.fr/hal-01018361>

Mean-field type controls are stochastic optimization problems involving statistical functions of the state and/or control such as their means and variance. This happens often in problems modeling risk for banks and energy optimization because typically some variables of the problem depend on the mean behavior of all actors, each optimizing the same cost function at their individual level.

Stochastic control is best analyzed by Dynamic Programming (DP) leading to the Hamilton-Jacobi-Bellman equation for the remaining cost $V(t)$, i.e. the optimization function from t to T , knowing that $x=x_t$, the state at t .

But here (see [1]) to apply DP we need to know the PDF of x_t not just its value at t and so the HJB equation contains derivatives with respect to measures.

We show in the conference that at the algorithmic level the extended DP is equivalent to standard calculus of variation applied to the deterministic problem derived from the stochastic one via the Fokker-Planck equation for the PDF of x_t . Thus the problem of finding the gradients and adjoints is solved.

We will illustrate the approach on 4 semi-academic mean-field type control problems

References

- 1 M. Lauriere and O. Pironneau: Dynamic programming for mean-field type control. C.R.A.S Serie I, 1–7, Oct. 2014.

3.23 The Edge Pushing Algorithm for Computing Sparse Hessians

Alex Pothen (Purdue University, US)

License  Creative Commons BY 3.0 Unported license
© Alex Pothen

We have revisited the Edge Pushing Algorithm for computing Hessians, proposed by Gower and Mello in 2012. We derive the algorithm using the notion of live variables from data flow analysis in compiler theory, showing that the algorithm maintains an invariant about the adjoints and Hessian matrix elements it computes at each step. We have implemented the algorithm to achieve correctness and efficiency in the context of computing Hessian matrices with the ADOL-C library for Algorithmic Differentiation. We have incorporated pre-accumulation in the computation to reduce the execution time. We provide rigorous complexity bounds for the algorithms and report execution times for a collection of test problems, including a mesh optimization problem. The results show that the Edge Pushing algorithm can be faster than the currently used algorithms (that compute compressed Hessian matrices via graph coloring) for some problems by one or two orders of magnitude, while also using less memory. Our implementation is available as open-source software, and will be included in a future release of ADOL-C.

References

- 1 R. M. Gower and M. P. Mello. *A new framework for Hessian Automatic Differentiation*. Optimization Methods and Software, vol. 27, 2012.

3.24 Hybrid Adjoint Approaches to Industrial Hydrodynamics

Thomas Rung (TU Hamburg-Harburg, DE)

License  Creative Commons BY 3.0 Unported license
© Thomas Rung

Joint work of Rung, Thomas; Arthur, Stück; Jörn, Kröger

The contribution reports the recent progress of the development of an adjoint Navier-Stokes method for incompressible flows [1] and its application to industrial hydrodynamics [2].

When attention is directed to gradient-based optimization, two approaches – the discrete and the continuous approach – are conceivable. The present research is derived from the continuous adjoint approach and aims at three crucial aspects for its industrial applicability, i. e. (a) accuracy and consistency, (b) numerical robustness and stability, (c) numerical efficiency and parallel performance.

The last aspect is a frequently addressed concern in conjunction with large-scale industrial applications and motivates the use of a continuous adjoint approach. As opposed to this, the first aspect is often identified as the origin of numerical problems and accuracy issues of the continuous adjoint methodology [3]. In order to support the duality between the discretised versions of the primal and the dual problem, a consistent discretization is derived for the building blocks of the adjoint system. The employed term-by-term strategy is based on the utilized primal, unstructured finite-volume discretization and their discrete adjoints obtained from the variation of the discrete Lagrangian. The strategy inheres continuous and discrete elements and is thus labeled hybrid adjoint approach. The approach supports the robustness of the algorithm and provides insight into an appropriate treatment of the adjoint coupling terms. Moreover, it facilitates a unified, discrete formulation of the adjoint wall-boundary condition and the related boundary-based sensitivity equation. Further details are found in [4].

References

- 1 A. Stück: Adjoint Navier-Stokes Methods for Hydrodynamic Shape Optimization. Dissertation, Hamburg University of Technology (TUHH), 2011.
- 2 A. Stück, J. Kröger, T. Rung: Adjoint-based hull design for wake optimization. *Ship Technol. Res.*, 58(1):34-44, 2011.
- 3 J. E. V. Peter and R. P. Dwight. Numerical sensitivity analysis for aerodynamic optimization: A survey of approaches. *Computers & Fluids*, 39(3):373–391, 2010.
- 4 A. Stück, T. Rung. Adjoint complement to viscous finite-volume pressure-correction methods. *Journal of Computational Physics*, 248:402–419, 2013.

3.25 Structure Exploitation, AD and the Continuous Problem

Stephan Schmidt (Universität Würzburg, DE)

License  Creative Commons BY 3.0 Unported license
© Stephan Schmidt

Hybridizing the discrete and continuous adjoint approach into a holistic, structure exploiting process is discussed. The potential advantages of this approach are exemplified within applications in the field of shape optimization as well as automatic code generation for FEM-problems such as provided by FEniCS.

Building upon previous studies [1], a potential hybrid Shape-AD tool could exploit the surface representation of the shape gradient, thereby automatically generating a primal/dual solver that can assemble a shape derivative by evaluating boundary quantities only. This could potentially lead to very fast and efficient code, automatically circumventing the necessity to consider mesh- or metric derivatives altogether by exploitation of the continuous problem structure.

Application examples where such a tool would be beneficial – starting by CFD problems and concluding with inverse design in acoustics and electromagnetism – are considered. The partial output of a preliminary semantic tool analyzing a shape optimization problem within the continuous Euler equations is shown as well.

References

- 1 C. Ilic, S. Schmidt, V. Schulz, and N. Gauger. Detailed aerodynamic shape optimization based on an adjoint method with shape derivatives. In M. Beckers, J. Lotz, V. Mosenkis, and U. Naumann, editors, *Fifth SIAM Workshop on Combinatorial Scientific Computing*, volume 2011-09, pages 8–10. Aachener Informatik-Berichte (AIB), 2011. ISSN 0935–3232.

3.26 Linking Adjoint Based Shape Optimization to Riemannian Geometry in Shape Spaces

Volker Schulz (Universität Trier, DE)

License © Creative Commons BY 3.0 Unported license
© Volker Schulz

Main reference V. Schulz, “A Riemannian View on Shape Optimization,” arXiv:1203.1493v2 [math.OC], 2014.

URL <http://arxiv.org/abs/1203.1493v2>

Shape optimization is a very active field of research with numerous applications of economic importance. Several examples from aerodynamics, acoustics and thermoelastics are used to illustrate this and to motivate the following more theoretical considerations. Although parametric geometry description (like CAD) are widely used in industry, they lead to high numerical costs for non-trivial geometry resolutions and pose severe limitations to the set of reachable shapes. The alternative avoiding these problems is the nonparametric approach which leaves all mesh nodes describing the geometry under investigation free for optimization and is based on the shape calculus. The current numerical state of the art in shape optimization based on the shape calculus is characterized by first order methods of steepest descent type and a general lack of second order methods. However, ideas from second order methods aiming a Newton-like strategies give rise to excellent preconditioners as demonstrated.

In this talk, a general framework based on differential geometric investigations is presented, which considers the set of admissible shapes as a Riemannian manifold and constructs Taylor series expansion and Newton methods similar to optimization ideas for finite dimensional matrix manifold. This novel approach, introduces a Riemannian shape Hessian as a Hessian formulation for second shape derivatives which, in contrast to the second shape derivative (which is so far historically but misleadingly named shape Hessian) possesses the properties which are expected from a Hessian: symmetry and provision of a Taylor series expansion. This approach is carried on to PDE constrained shape optimization and develops a novel sequential quadratic programming framework for shape optimization based on the shape calculus, where the linear-quadratic subproblems to be solved in each nonlinear iteration have the structure of usual optimal control problems and are thus accessible to the wealth of efficient methods developed for this problem class like, e. g., multigrid optimization methods.

3.27 Discrete Adjoint Optimization for OpenFOAM

Markus Towara (RWTH Aachen University, DE)

License © Creative Commons BY 3.0 Unported license
© Markus Towara

Joint work of Towara, Markus; Sen, Arindam; Naumann, Uwe

Main reference M. Towara, U. Naumann, “A Discrete Adjoint Model for OpenFOAM,” in Proc. of the 2013 Int’l Conf. on Computational Science, Procedia Computer Science, Vol. 18, pp. 429–438, 2013.

URL <http://dx.doi.org/10.1016/j.procs.2013.05.206>

OpenFOAM is an Open-Source CFD Simulation Tool with a wide range of applications and a strongly growing user base in both academia and industry. The source code is available in C++. The application of the adjoint model is a common approach for high dimensional optimization problems, however often a continuous approach instead of a discrete one is used. Codes which generate adjoint sensitivity information using the discrete approach are usually generated by Algorithmic Differentiation[1], either by source code transformation or operator

overloading, thus differentiating the code on a per statement level instead of deriving and discretizing a set of adjoint equations as with the continuous approach. We introduced a discrete adjoint version of OpenFOAM using the operator overloading tool `dco/c++` in order to generate derivatives and to apply optimization (with a focus on topology optimization)[2]. A discrete adjoint implementation in general yields a significant overhead to the passive evaluation in computation time and more importantly required memory. (Intermediate values from the whole computational history have to be stored in order to evaluate the derivatives). Our work focuses on how we managed to significantly reduce the memory requirements of said discrete adjoint OpenFOAM version, i. e. by applying analytical knowledge about the iterative linear solvers used to solve the underlying partial differential equations, thus eliminating the need to store intermediate values generated during the solution process of the linearized equations[3]. This treatment yields a significant improvement in both runtime and memory usage and also eliminates the dependency of the memory usage on the number of linear solver iterations.

References

- 1 A. Griewank and A. Walther, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, SIAM, 2008
- 2 M. Towara and U. Naumann, *A Discrete Adjoint Model for OpenFOAM*, *Procedia Computer Science* Vol. 18, 2013
- 3 M. Giles, *Collected Matrix Derivative Results for Forward and Reverse Mode Algorithmic Differentiation*, *Advances in Automatic Differentiation*, 2008

3.28 Convergence of discrete adjoints for flows with shocks

Stefan Ulbrich (TU Darmstadt, DE)

License © Creative Commons BY 3.0 Unported license
© Stefan Ulbrich

Joint work of Giles, Mike; Ulbrich, Stefan

Main reference M. Giles, S. Ulbrich, “Convergence of linearized and adjoint approximations for discontinuous solutions of conservation laws. Part 1: Linearized approximations and linearized output functionals,” *SIAM J. on Numerical Analysis*, 48(3):882–904, 2010.

URL <http://dx.doi.org/10.1137/080727464>

Main reference M. Giles, S. Ulbrich, “Convergence of linearized and adjoint approximations for discontinuous solutions of conservation laws. Part 2: Adjoint Approximations and Extensions,” *SIAM J. on Numerical Analysis*, 48(3):905–921, 2010.

URL <http://dx.doi.org/10.1137/09078078X>

We analyze the convergence of discrete adjoint approximations for optimal control problems governed by an unsteady one-dimensional hyperbolic conservation law with a convex flux function. A simple modified Lax-Friedrichs discretization is used on a uniform grid that has a numerical viscosity of $O(h^\alpha)$, $2/3 < \alpha < 1$, and we consider a tracking type objective function at the end time. The control are the initial data. It is known that such tracking type objective functions are differentiable with respect to the initial control also in the case of shocks and that an adjoint based representation of the reduced gradient of the objective can be obtained [1, 4, 5]. We show that the discrete adjoint scheme converges pointwise almost everywhere and uniformly outside of any neighborhood of the extreme backward characteristics emanating from shocks, see [2, 3]. A key point is that the numerical smoothing increases the number of points across the nonlinear discontinuity as the grid is refined. Hence, the discrete adjoint leads to a convergent representation of the reduced objective gradient. We sketch the proof idea of [2, 3] which is based on an asymptotic expansion with respect

to the viscosity parameter in an inner region around the shock and an outer region. In addition, we present numerical results illustrating the asymptotic behavior which is analyzed. Finally, we illustrate that a numerical viscosity of $O(h^\alpha)$, $2/3 < \alpha < 1$ is necessary to obtain convergence of the discrete adjoint if a shock is present in the region, where the objective function is evaluated.

References

- 1 Michael B. Giles, *Discrete adjoint approximations with shocks*, in *Hyperbolic Problems: Theory, Numerics, Applications*, T. Hou and E. Tadmor, eds., Springer-Verlag, New York, 2003
- 2 Mike Giles and Stefan Ulbrich, *Convergence of linearized and adjoint approximations for discontinuous solutions of conservation laws. Part 1: Linearized approximations and linearized output functionals*, *SIAM J. Numer. Anal.*, 48 (2010), pp. 882–904.
- 3 Mike Giles and Stefan Ulbrich, *Convergence of linearized and adjoint approximations for discontinuous solutions of conservation laws. Part 2: Adjoint approximations and extensions*, *SIAM J. Numer. Anal.*, 48 (2010), pp. 905–921.
- 4 Stefan Ulbrich, *A sensitivity and adjoint calculus for discontinuous solutions of hyperbolic conservation laws with source terms*, *SIAM J. Control Optim.*, 41 (2002), pp. 740–797
- 5 Stefan Ulbrich, *Adjoint-based derivative computations for the optimal control of discontinuous solutions of hyperbolic conservation laws*, *Systems Control Lett.*, 48 (2003), pp. 313–328.

3.29 White box adjoining of a radiative transport model and its use for ill-posed inverse problems

Joern Ungermann (*Forschungszentrum Jülich, DE*)

License © Creative Commons BY 3.0 Unported license
© Joern Ungermann

Main reference J. Ungermann, “Tomographic reconstruction of atmospheric volumes from infrared limb-imager measurements,” Dissertation, Wuppertal University, Schriften des Forschungszentrums Jülich, Series Energie & Umwelt, Vol. 106, 153 pages, 2011.

URL <http://hdl.handle.net/2128/4385>

The Gimballed Limb Radiance Imager of the Atmosphere (GLORIA) is a newly developed unique atmospheric sounder that combines for the first time a classical Fourier transform spectrometer (FTS) with a 2-D detector array. Imaging allows the spatial sampling to be improved by up to an order of magnitude when compared to a conventional limb scanning instrument. GLORIA is designed to operate on various high altitude aircrafts.

Its unique scanning scheme and data acquisition rate allows for the first time the tomographic measurement of large air volumes about a thousand kilometers across. Reconstructing 3-D volumes from the measured infrared spectra thus poses a large-scale inverse problem, which requires a highly optimized forward model and inversion scheme.

Implementing an adjoint version using the dco tool suite of STCE, RWTH Aachen University, was the first step towards that goal. Separating the forward model internally into a linear map and a set of functions with one-dimensional output allowed to construct the full Jacobian matrix of the forward model by just a single execution without checkpointing. The performance could be further increased by a factor of two by manually computing the Jacobians of often-called subroutines and directly inserting these values into the tape, thereby drastically reducing tape-size.

Solving the inverse problem requires the minimization of a cost function composed of a term describing the agreement between measurements and simulated measurements for a given atmospheric state on the one hand and a regularizing term on the other hand. Having available the full Jacobian matrix of the forward model allows the efficient implementation of Quasi-Newton methods to minimize the cost function. These methods approximate the Hessian of the cost function by neglecting the Hessian of the forward model. The quadratic convergence of these methods requires usually less than 10 iterations for sufficient results and thereby only as many evaluations of the Jacobian matrix of the forward model. The numerical properties of the minimizer were greatly increased by providing an approximate Jacobian preconditioner for the Hessian of the cost function, which can be straightforwardly approximated given the available matrices. Lastly, each Quasi-Newton iteration requires the solution to a linear equation system, which can be produced matrix-free using conjugate-gradients. The regularizing properties of the conjugate-gradient scheme are used to implement a trust-region method, where intermediate solutions of increasing accuracy are stored along the computation of the precise solution and used in a back-tracking fashion to prevent the costly solving of additional linear equation systems.

3.30 Time-minimal Checkpointing

Andrea Walther (Universität Paderborn, DE)

License  Creative Commons BY 3.0 Unported license
© Andrea Walther

Joint work of Walther, Andrea; Griewank, Andreas

For adjoint calculations, parameter estimation, and similar purposes one may need to reverse the execution of a computer program. The simplest option is to record a complete execution log and to read it backwards as required. This approach may require massive amounts of storage. Instead one may generate the execution log piecewise by restarting the “forward” calculation repeatedly from suitably placed checkpoints.

The basic structure of the resulting reversal schedules is illustrated. Various strategies are analyzed with respect to the resulting temporal and spatial complexity on serial and parallel machines. For serial machines known optimal compromises between operations count and memory requirement are explained.

For program execution reversal on multi-processors the new challenges and demands on an optimal reversal schedule are described. We present parallel reversal schedules that are provably optimal with regards to the number of concurrent processes and the total amount of memory required. More details on this time-minimal checkpointing approach can be found in [1, 2].

References

- 1 Andrea Walther: Program Reversal Schedules for Single- and Multi-processor Machines. Promotion, TU Dresden, 2000.
- 2 Andrea Walther: Bounding the number of processes and checkpoints needed in time-minimal parallel reversal schedules. *Computing* 73:135–154 (2004).

3.31 Adjoining Chaos

Qiqi Wang (MIT – Cambridge, US)

License © Creative Commons BY 3.0 Unported license
© Qiqi Wang

Joint work of Blonigan, Patrick; Gomez, Steven; Hu, Rui; Gauger, Nico; Diskin, Boris; Nielson, Eric; Farrell, Patrick

Main reference Q. Wang, R. Hu, P. Blonigan, “Least Squares Shadowing sensitivity analysis of chaotic limit cycle oscillations,” *Journal of Computational Physics*, 267(June 2014):210–224, 2014; pre-print available as arXiv:1204.0159v7 [physics.comp-ph].

URL <http://dx.doi.org/10.1016/j.jcp.2014.03.002>

URL <http://arxiv.org/abs/1204.0159v7>

The adjoint method, among other sensitivity analysis methods, can fail in chaotic dynamical systems. The result from these methods can be too large, often by orders of magnitude, when the result is the derivative of a long time averaged quantity. This failure is known to be caused by ill-conditioned initial value problems. This paper overcomes this failure by replacing the initial value problem with the well-conditioned least squares shadowing (LSS) problem. The LSS problem is then linearized in our sensitivity analysis algorithm, which computes a derivative that converges to the derivative of the infinitely long time average. We demonstrate our algorithm in several dynamical systems exhibiting both periodic and chaotic oscillations.

Participants

- Christian Bischof
TU Darmstadt, DE
- Luise Blank
Universität Regensburg, DE
- David Bommers
INRIA Sophia Antipolis –
Méditerranée, FR
- Martin Bücker
Universität Jena, DE
- Bruce Christianson
University of Hertfordshire, GB
- Mike Dewar
NAG Ltd. – Oxford, GB
- Boris Diskin
National Institute of Aerospace –
Hampton, US
- Patrick Farrell
University of Oxford, GB
- Christian Frey
DLR – Köln, DE
- Nicolas R. Gauger
TU Kaiserslautern, DE
- Andreas Griewank
HU Berlin, DE
- Stefanie Günther
TU Kaiserslautern, DE
- Ralf Hannemann-Tamas
Univ. of Science & Technology –
Trondheim, NO
- Laurent Hascoet
INRIA Sophia Antipolis –
Méditerranée, FR
- Patrick Heimbach
MIT – Cambridge, US
- Paul D. Hovland
Argonne National Laboratory, US
- Barbara Kaltenbacher
Universität Klagenfurt, AT
- Peter Korn
MPI für Meteorologie –
Hamburg, DE
- Drosos Kourounis
University of Lugano, CH
- Ben Lenser
HU Berlin, DE
- Johannes Lotz
RWTH Aachen, DE
- Marcus Meyer
Rolls-Royce –
Blankenfelde-Mahlow, DE
- Viktor Mosenkis
RWTH Aachen, DE
- Jens-Dominik Mueller
Queen Mary University of
London, GB
- Uwe Naumann
RWTH Aachen, DE
- Eric Nielsen
NASA Langley ASDC –
Hampton, US
- Jacques Peter
ONERA – Châtillon, FR
- Olivier Pironneau
UPMC – Paris, FR
- Alex Pothen
Purdue University, US
- Thomas Rung
TU Hamburg-Harburg, DE
- Ekkehard Sachs
Universität Trier, DE
- Max Sagebaum
TU Kaiserslautern, DE
- Stephan Schmidt
Universität Würzburg, DE
- Volker Schulz
Universität Trier, DE
- Markus Towara
RWTH Aachen, DE
- Stefan Ulbrich
TU Darmstadt, DE
- Jörn Ungermann
Forschungszentrum Jülich, DE
- Jean Utke
Allstate – Northbrook, US
- Andrea Walther
Universität Paderborn, DE
- Qiqi Wang
MIT – Cambridge, US

