Contents

Householder Committees 16

Plenary speakers 18

Speakers 19

A framework for second-order eigenvector centralities and clustering coefficients 24
Francesca Arrigo, Desmond J. Higham and Francesco Tudisco

Solving matrix equations associated with random walks in the quarter plane: theoretical and algorithmic analysis 27
Dario A. Bini, Stefano Massei, Beatrice Meini, Jie Meng, Leonardo Robol

Some Distance Problems for Matrix Polynomials via Block Toeplitz Matrices 30
Shreemayee Bora, and Biswajit Das

Iterative Refinement in Three Precisions 32
Erin C. Carson, Nicholas J. Higham, and Srikara Pranesh

Asynchronous Iterative Methods 35
Edmond Chow

Efficient Krylov Subspace Methods for Uncertainty Quantification in Large Linear Inverse Problems 38
Julianne Chung, Katrina Petroske, and Arvind K. Saibaba

Sketching-based Algorithms for Ridge Regression and Applications 41
Petros Drineas

Use of Partitioning in Solving Sparse Linear Equations on Heterogeneous High Performance Computers 44
Iain Duff

Krylov Methods for Low-Rank Regularization 47
Silvia Gazzola, Chang Meng, and James G. Nagy

Contour Integral Methods for Nonlinear Eigenvalue Problems: A Systems Theoretic Approach 50
Michael C. Brennan, Mark Embree, and Serkan Gugercin
Probabilistic Numerical Linear Solvers
Jon Cockayne, Ilse C.F. Ipsen, Chris J. Oates and Tim Reid

Numerical Differential Geometry
Tingran Gao, Lek-Heng Lim, and Ke Ye

Block Krylov Subspace Methods for Functions of Matrices II: Modified block FOM
Andreas Frommer, Kathryn Lund, and Daniel B. Szyld

Sharp Probabilistic Rounding Error Analysis for Large Scale, Low Precision Scientific Computing
Nicholas J. Higham and Theo Mary

Spurious Eigenvalues of the Linearization of Nonlinear Eigenvalue Problems
Karl Meerbergen

An Interpolatory Subspace Framework for Nonlinear Eigenvalue Problems
Emre Mengi

Towards Randomized Eigensolvers for Exascale Computing
Agnieszka Miedlar

Partial Smoothness of the Numerical Radius
Adrian S. Lewis and Michael L. Overton

Approximation of the Inverse Hessian in Large-Scale Variational Data Assimilation Problems
Alison Ramage

Randomized Discrete Empirical Interpolation Method for Nonlinear Model Reduction
Arvind K. Saibaba

A Rational Even-Ira Algorithm for the solution of alternating polynomial Eigenvalue Problems
Heike Faßbender, Peter Benner, Philip Saltenberger

Spectral estimates for saddle point matrices arising in numerical weather prediction
Ieva Daužickaitė, Amos S. Lawless, Jennifer A. Scott, and Peter Jan van Leeuwen

Graph-based PDEs: Laplacians, eigeninformation, and semi-supervised learning
Resolvent-based techniques for computing the discrete and continuous spectrum of differential operators

Matthew Colbrook, Andrew Horning, and Alex Townsend

Learning from Network Data with Nonlinear Eigenvectors

Francesco Tudisco, and Desmond J. Higham

Analytical Low-rank Compression via Proxy Point Selection

Jianlin Xia, Xin Ye, and Lexing Ying

Communication-avoiding TT-tensor orthogonalization and rounding procedures

Hussam Al Daas, Grey Ballard, and Peter Benner

A Localization Preserving Preconditioner for Nonlinear Eigenvector Problems under Disorder

Robert Altmann, and Daniel Peterseim

Reduction of Systems with Polynomial Nonlinearities in the Loewner Framework

Athanasios C. Antoulas, I. V. Gosea, and M. Heinkenschloss

Interpolation spaces: a numerical linear algebra perspective

Mario Arioli

Randomization for the Efficient computation of Reduced Order Models

Selin Aslan, Eric de Sturler and Serkan Gugercin

Meeting the challenges of computing many eigenpairs

Zhaojun Bai, Jack Dongarra, Chao-Ping Lin, Ding Lu and Ichitaro Yamazaki

Block Modified Gram-Schmidt Algorithms and their Analysis

Jesse L. Barlow

Reduction of Time-Periodic Systems: the Floquet-Arnoldi Decomposition

Christopher Beattie

Complex block Jacobi method with application to PGEP

Erna Begović Kovač and Vjeran Hari

Greedy Low-rank Connectome Regression
Peter Benner, Sergey Dolgov, Kameron Decker Harris, and Patrick Kürschner

Hodge Laplacians and Random Walks on Simplicial Complexes 113

Michael T. Schaub, Austin R. Benson, Paul Horn, Gabor Lippner, and Ali Jadbabaie

Nonlocal Diffusion on Networks via Fractional Graph Laplacians: Algorithms and Applications 116

Michele Benzi

Efficient preconditioners for MINRES in Convex Constrained Optimization 117

Luca Bergamaschi, Jacek Gondzio, Ángeles Martínez, John Pearson and Spyridon

The Sad Truth about Linear Algebra Software 119

Henrik Barthels, Christos Psarras, Paolo Bientinesi

On the Characterization of Linear GMRES Convergence and its Stability with respect to Compact Perturbations 121

Jan Blechta

Low Complexity Computation of the Pseudo-Inverse of a Digraph Laplacian 124

Daniel Boley

Block smoothers in multigrid methods for structured matrices 126

Matthias Bolten

A Preconditioner based on Sparsified Nested Dissection and Low-Rank Approximation 128

Erik G. Boman, Leopold Cambier, Chao Chen, Eric Darve, Siva Rajamanickam, Ray S. Tuminaro

Compact Representations of Structured Quasi-Newton Update Matrices 130

Johannes J. Brust, Sven Leyffer, and Cosmin G. Petra

Regularization parameter selection rules for unconstrained and constrained $\ell^p - \ell^q$ minimization 132

Alessandro Bucci, Mirjeta Pasha, and Lothar Reichel

Sparse Matrices Beyond Solvers: Graphs, Biology, and Machine Learning 135

Aydin Buluç

Fast Evaluation and Approximation of the Gauss-Newton Hessian Matrix for the Multilayer Perceptron 138
Chao Chen, Severin Reiz, Chenhan Yu, Hans-Joachim Bungartz, and George Biros

Analyzing the Effects of Local Roundoff Error on Predict-and-Recompute Conjugate Gradient Variants

Tyler Chen, Erin Carson, Anne Greenbaum

Sampled Limited Memory Methods for Massive Least Squares Problems

Julianne Chung, Matthias Chung, J. Tanner Slagel, and Luis Tenorio

On a Spectral Method for “Bipartizing” a Network and Detecting a Large Anti-Community

Anna Concas, Silvia Noschese, Lothar Reichel and Giuseppe Rodriguez

Cross Approximation, Column Subset Selection, and Maximum Volume Submatrices

Alice Cortinovis, Daniel Kressner, and Stefano Massei

Uniform Invariant Subspace Perturbation Theory

Anil Damle, and Yuekai Sun

Multiparameter eigenvalue problems

Bart De Moor

Krylov Recycling for Iterative Hybrid Projection Methods with Applications to Streaming Data

Julianne M. Chung, Eric de Sturler, and Jiahua Jiang

SVD Parameterization for Deep Neural Networks

Inderjit S. Dhillon, and Jiong Zhang

Versal Deformations in Numerical Linear Algebra

Andrii Dmytryshyn

Multi-linear algebra for approximating high dimensional functions in Bayesian inverse problems

Sergey Dolgov, Karim Anaya-Izquierdo, Colin Fox, Robert Scheichl

Backward stability of rational eigenvalue problems solved via linearization

Froilán M. Dopico, María C. Quintana, and Paul Van Dooren

New computational tools for Koopman spectral analysis of nonlinear dynamical systems
Embedding properties of network realizations of dissipative reduced order models

On the Convergence of Two-Level AMG with aggregation-based coarsening relying on weighted matching

Partitioning Large and Sparse Tensors using Best Low Rank Decompositions

Compact Reduced-Basis Methods for Time-Dependent Partial Differential Equations

Iteratively Corrected Rational Interpolation for Nonlinear Eigenvalue Problems

Fast Generation of Extreme-Scale Matrices with Pre-Assigned 2-Norm Condition Number

Higher-order Markov Chains, Tensor Equations, and Ergodicity Coefficients

Inexact Methods for the Low-Rank Solution to Large-Scale Lyapunov Equations

Krylov type methods exploiting the quadratic numerical range

Multigrid Methods for Block-Toeplitz linear systems: Convergence Analysis and Applications

Balanced truncation for linear switched systems
Automatic Generation of Minimal and Reduced Models for Structured Parametric Dynamical Systems 199

Peter Benner, Pawan Goyal, and Igor Pontes Duff

Parallel Variants of the Conjugate Gradient Algorithm and their Performance in Finite Precision Arithmetic 201

Anne Greenbaum, Hexuan Liu, and Tyler Chen

A Scalable Approximate Inverse Block Preconditioner for an Incompressible Magnetohydrodynamics Model Problem 203

Michael Wathen and Chen Greif

A multilevel Schwarz preconditioner based on a hierarchy of robust coarse spaces 205


Hyperbolic problems for metric graphs – tools for analyzing vibrations of frame like structures. 208

Luka Grubišić, Josip Tambača, Matko Ljulj and Voker Mehrmann

Computing the approximate common divisor of polynomials 210

Antonio Fazzi, Nicola Guglielmi, and Ivan Markovsky

Sharp 2-Norm Error Bounds for the Conjugate Gradient Method and LSQR 212

Eric Hallman

Algorithms with automatic result verification for the matrix exponential 214

Andreas Frommer, and Behnam Hashemi

Cluster Gauss-Newton method for finding Multiple Global Minimizers of Nonlinear Least Squares problems - with application to Parameter Estimation of Pharmacokinetic Models 216

Yasunori Aoki, Ken Hayami, Kota Toshimoto, and Yuichi Sugiyama

Extracting relevant data from linear data fitting problems via generalized Krylov subspace methods 219

Iveta Hnětynková, Martin Plešinger, and Jana Žaková

NEP-PACK: A Julia package for nonlinear eigenproblems 222

Elias Jarlebring, Max Bennedich, Giampaolo Mele, Emil Ringh, Parikshit Upadhyaya

Hybrid Projection Methods for Large-scale Inverse Problems with Mixed Gaussian Priors 225
Taewon Cho, Julianne Chung, Jiahua Jiang

Parallel Numerical Linear Algebra for Future Extreme Scale Systems: Challenges and Highlights of NLAFET

Bo Kågström, and the NLAFET Team

Don’t Matricize, Tensorize: Tensor-tensor Products for Optimal Representation and Compression

Misha E. Kilmer, Lior Horesh, Haim Avron and Elizabeth Newman

Order reduction methods for solving large-scale differential matrix Riccati equations

Gerhard Kirsten, and Valeria Simoncini

A multivariate Crouzeix-Palencia bound and its applications

Daniel Kressner

Admissible and attainable convergence behavior of block Arnoldi and GMRES

Marie Kubinová, and Kirk M. Soodhalter

Numerical linear and multilinear algebra methods for systems of polynomials

Patrick Kürschner, Lieven De Lathauwer

Delaying the second orthogonalization in Classical Gram-Schmidt with Reorthogonalization to obtain a one-synchronization Gram-Schmidt algorithm for QR factorizations and Arnoldi expansions. Application to GMRES and Krylov-Schur algorithms.

Julien Langou, Kasia Świrydowicz, Stephen Thomas, Daniel Bielich

Structure Analysis of Three Dimensional Maxwell Eigenvalue Problems

Tsung-Ming Huang, Tiexiang Li, Wen-Wei Lin and Heng Tian

A communication-avoiding 3D sparse triangular solver

Piyush Sao, Ramakrishnan Kannanm, Xiaoye S. Li, Richard Vuduc

The Forsythe Conjecture

Jörg Liesen

Quantum Linear System Solver

Dong An, Lin Lin

 Polynomial Preconditioning with the GMRES Polynomial

 Polynomial Preconditioning with the GMRES Polynomial
Subspace Acceleration for Convex Optimization Over the Numerical Range 256
\textit{Daniel Kressner, Ding Lu, and Bart Vandereycken}

Computing the Eigenvalues of Large Hadamard Matrix Functions using Low Rank Approximations 257
\textit{Omar De La Cruz Cabrera, Thomas Mach, and Lothar Reichel}

A Natural Vector Space Setting for Block Minimal Basis Pencils 259
\textit{D. Steven Mackey, Javier Pérez, and Philip Saltenberger}

Fast sampling of parameterised Gaussian random fields via low-rank approximation 262
\textit{D. Kressner, J. Latz, S. Massei, and E. Ullmann}

Computing Eigenvectors in Floating Point Arithmetic 264
\textit{Teresa Laudadio, Nicola Mastronardi, and Paul Van Dooren}

Multigrid Strategies for both Isotropic and Anisotropic Fractional Diffusion Problems 266
\textit{Mehdi Dehghan, Marco Donatelli, Rolf Krause, Mariarosa Mazza, Hamid Moghaderi, Ken Trotti}

Robust representations of passive systems in port-Hamiltonian form 269
\textit{Volker Mehrmann, and Paul Van Dooren}

The infinite Lanczos method for symmetric nonlinear eigenvalue problems 270
\textit{Giampaolo Mele}

On prescribing the convergence behaviour of the Conjugate Gradient algorithm 273
\textit{Gerard Meurant}

Computing the Kreiss Constant of a Matrix 275
\textit{Tim Mitchell}

Deflation for Krylov Solution of Linear Equations 278
\textit{Ron Morgan}

Verifying partial eigenvalues of Hermitian generalized eigenproblems using contour integrals 281
\textit{Akira Imakura, Keiichi Morikuni, and Akitoshi Takayasu}
An algorithm to numerically compute the Clebsch-Gordan coefficients of unitary representations of compact groups

A. Ibort, A. López-Yela, and J. Moro

Task-based Algorithms and Software for Solving Dense Nonsymmetric Eigenvalue Problems

Mirko Myllykoski

Deflation and Multigrid: Similarities and Differences

Reinhard Nabben

Partial Nonnegative Matrix Factorization in Fluorescence Microscopy

James Nagy, and Blair Rossetti

Symmetric eigenvalue perturbation in the tangent space

Yuji Nakatsukasa

Towards a robust eigensolver for large sparse nonlinear eigenproblems

Stefan Güttel, Gian Maria Negri Porzio, and Françoise Tisseur

Efficient Training of Neural Networks via Variable Projection

Elizabeth Newman, Joseph Hart, Bart van Bloemen Waanders, and Lars Ruthotto

Fast implementation of the traveling-salesman-problem method for reordering columns within supernodes in sparse Cholesky factorization

Mathias Jacquelin, Esmond Ng, and Barry Peyton

Tensor Completion and its Applications

Michael K. Ng

Treating Ensemble Sample Error Covariances in Data Assimilation

Nancy K Nichols, Polly J Smith and Amos S. Lawless

Inertia laws and localization of real eigenvalues for generalized indefinite eigenvalue problems

Yuji Nakatsukasa, and Vanni Noferini

The SVD and JCF reveal the structure in LOSS OF ORTHOGONALITY

Chris Paige

Matrix Equation Methods for Certain Evolutionary Partial Differential Equations
From a posteriori algebraic error estimator to multilevel iterative solver with $p$-robust behavior

Ani Mirači, Jan Papež, Martin Vohralík

Recent Advances in the Preconditioned Iterative Solution of Time-Dependent PDE-Constrained Optimization Problems

John W. Pearson

Preconditioners for Symmetrized Toeplitz and Multilevel Toeplitz Problems

Jennifer Pestana

Recent development of the core problem theory in the context of the total least squares minimization

Iveta Hnětynková, Martin Plešinger, and Jana Žáková

Solving Singular Generalized Eigenvalue Problems by Rank-Completing Perturbations

Michiel E. Hochstenbach, Christian Mehl, and Bor Plestenjak

Balanced Truncation for Linear System with Quadratic Output: Theory, Error Bounds and Numerics

Peter Benner, Pawan Goyal, and Igor Pontes Duff

Best $k$-layer neural network approximations

Lek-Heng Lim, Mateusz Michalek, and Yang Qi

Novel techniques for exploiting low-rank structures in nonlinear eigenvalue problems

Javier Pérez and María C. Quintana

Krylov meets Bregman

Alessandro Buccini, Mirjeta Pasha, and Lothar Reichel

Improving Krylov subspace approximations for solution of ill-posed inverse problems: oversampling and truncation

Anthony Helmstetter, Rosemary Renaut

Nonlinearizing two-parameter eigenvalue problems

Emil Ringh, and Elias Jarlebring
Rational Krylov for Stieltjes matrix functions: convergence and pole selection 342
  
  Stefano Massei, and Leonardo Robol

Numerical Homotopy Methods for Multiparameter Eigenvalue Problems 344
  
  Lek-Heng Lim, Jose Israel Rodriguez, and Yiling You

Spectral Properties of Doubly-Stochastic Scaling to Uncover Hidden Block Structure 347
  
  Iain S. Duff, P. A. Knight, L. le Gorrec, S. Mouysset and D. Ruiz

Iteratively Reweighted FGMRES for Sparse Reconstruction 350
  
  Silvia Gazzola, James G. Nagy, and Malena Sabaté Landman

SSAI: A symmetric sparse approximate inverse preconditioner for PCG 352
  
  Shaked Regev and Michael A. Saunders

Computing the closest real normal matrix and the normal completion 354
  
  Nicola Guglielmi and Carmela Scalone

Recent progress on nonlinear eigenvalue problems with algebraic branch points 356
  
  Zhaojun Bai, Weiguo Gao, Xin Huang, Meiyue Shao, Roel Van Beeumen, and Chao Yang

On computing the STRUCTURED DISTANCE TO SINGULARITY for matrix pencils using optimization 358
  
  Punit Sharma

Improved Algorithms and Parallel Software for CP Decomposition 360
  
  Linjian Ma, Navjot Singh, and Edgar Solomonik

A unifying framework for recycling-based iterative methods 363
  
  Kirk M. Soodhalter, Eric de Sturler, and Misha Kilmer

Spectral information in second order differential operators and cost of iterative computations 366
  
  E. Carson, T. Gergelits, K.-A. Mardal, B. F. Nielsen, and Z. Strakoš

Simultaneous Diagonalization of Nearly Commuting Hermitian Matrices: Do-one-then-do-the-other 368
  
  Brian Sutton

Asynchronous Schwarz Methods: Theory and Computations 370
Daniel B. Szyld

The Conditioning of Least Squares Problems in Variational Data Assimilation

Jemima M. Tabeart, Sarah L. Dance, Amos S. Lawless, Nancy K. Nichols and Joanne A. Waller

Half-radial matrices and Crouzeix’s conjecture

Iveta Hnětynková, Petr Tichý

An Algorithm for Dense Nonlinear Eigenvalue Problems

Gian Maria Negri Porzio and Françoise Tisseur

Null-space approach for solving symmetric augmented systems

Jennifer Scott, Miroslav Tůma

A Flexible Power Method for Infinite Dimensional Tensor Eigenvalue Problems

Roel Van Beeumen, Lana Periša, Daniel Kressner, and Chao Yang

Detecting roles in large graphs

Melissa Marchand, Kyle Gallivan, Wen Huang, and Paul Van Dooren

The influence of condition numbers on the convergence of rank-constrained optimization problems

André Uschmajew and Bart Vandereycken

Low-rank plus Sparse Matrices: Ill-posedness and Recovery

Jared Tanner and Simon Vary

Overlapping Community Detection via Eigendecomposition

Jimit Majmudar, Stephen Vavasis

Algebraic and Optimization-based Algorithms for Decomposing Tensors into Block Terms

Nico Vervliet, Ignat Domanov, Lieven De Lathauwer

$\mathcal{H}_\infty$-Control of Large-Scale Systems

Nicat Aliyev, Peter Benner, Emre Mengi, Tim Mitchell, Paul Schwerdtner, and Matthias Voigt

Preconditioning for nonsymmetric problems

Andy Wathen
Solving the Matrix Eigenvalue Problem: chasing bulges, cores, or poles
David S. Watkins

Fast randomized iterative numerical linear algebra and application to quantum chemistry
Timothy Berkelbach, Samuel Greene, Lek-Heng Lim, Jonathan Weare, and Robert Webber

Distance Problems for Dissipative Hamiltonian Systems
Christian Mehl, Volker Mehrmann, Michał Wojtylak

Accelerating Parallel Hierarchical Matrix-Vector Products via Data-Driven Sampling
Yuanzhe Xi, Difeng Cai, Lucas Erlandson and Edmond Chow

Interpolative Decomposition via Proxy Points for Kernel Matrices
Xin Xing, and Edmond Chow

Krylov subspace methods for the trust-region subproblem and beyond
Lei-Hong Zhang, Ren-Cang Li, and Chungen Shen

Linear algebraic approach to inverse scattering problems
Liliana Borcea, Vladimir Druskin, Mikhail Zaslavsky, Alexander Mamonov and Jörn Zimmerling

Computing the restricted singular value decomposition with improved robustness
Ian Zwaan
Householder Committee

- Zhaojun Bai, University of California at Davis, United States of America
- David Bindel, Cornell University, United States of America
- James Demmel, University of California at Berkeley, United States of America
- Zlatko Drmač, University of Zagreb, Croatia
- Heike Faßbender (chair and SIAM representative), Technical University of Braunschweig, Germany
- Sherry Li, Lawrence Berkeley National Laboratory, United States of America
- Volker Mehrmann, Technical University of Berlin, Germany
- James Nagy, Emory University, United States of America
- Valeria Simoncini, University of Bologna, Italy
- Andrew Wathen, Oxford University, United Kingdom

Local Householder Committee

- Dario A. Bini, University of Pisa
- Fasma Diele, Istituto per le Applicazioni del Calcolo ”M. Picone” - CNR, Bari
- Beatrice Meini, University of Pisa
- Teresa Laudadio, Istituto per le Applicazioni del Calcolo ”M. Picone” - CNR, Bari
- Carmela Marangi, Istituto per le Applicazioni del Calcolo ”M. Picone” - CNR, Bari
- Nicola Mastronardi (chair), Istituto per le Applicazioni del Calcolo ”M. Picone” - CNR, Bari
- Stefano Serra Capizzano, University of Insubria
- Valeria Simoncini, University of Bologna

Householder Prize Committee

- Inderjit Dhillon, UT Austin and Amazon, United States of America
- Alan Edelman, MIT, United States of America
• Mark Embree, Virginia Tech, United States of America
• Melina Freitag, University of Bath, United Kingdom
• Andreas Frommer, University of Wuppertal, Germany
• Françoise Tisseur, University of Manchester, United Kingdom
• Stephen Vavasis, University of Waterloo, Canada
Plenary speakers

- Francesca Arrigo, University of Strathclyde, United Kingdom
- Dario Andrea Bini, Department of Mathematics, University of Pisa, Italy
- Shreemayee Bora, Indian institute of Technology Guwahati, India
- Erin C. Carson, Charles University, Czech Republic
- Edmond Chow, Georgia Institute of Technology, United States
- Julianne Chung, Virginia Tech, United States
- Petros Drineas, Purdue University, United States
- Iain Duff, STFC Rutherford Appleton Laboratory
- Silvia Gazzola, University of Bath, United Kingdom
- Serkan Gugercin, Virginia Tech, United States
- Ilse C.F. Ipsen, North Carolina State University, United States
- Lek-Heng Lim, University of Chicago, United States
- Kathryn Lund, Charles University, Czech Republic
- Theo Mary, CNRS, France
- Karl Meerbergen, KULeuven, Belgium
- Emre Mengi, Koc University, Istanbul, Turkey
- Agnieszka Miedlar, University of Kansas, USA
- Michael Overton, Courant Institute, NYU, United States
- Alison Ramage, University of Strathclyde, Scotland
- Arvind Krishna Saibaba, North Carolina State University, United States
- Philip Saltenberger, TU Braunschweig, Germany
- Jennifer Scott, University of Reading and STFC Rutherford Appleton Laboratory, United Kingdom
- Martin Stoll, TU Chemnitz, Germany
- Alex Townsend, Cornell University, United States
- Francesco Tudisco, Gran Sasso Science Institute, Italy
- Jianlin Xia, Purdue University, United States
Speakers

- Hussam Al Daas, Max Planck Institute for Dynamics of Complex Technical Systems, Germany
- Robert Altmann, University of Augsburg, Germany
- Athanasios Antoulas, Rice University and Max-Planck Institute, United States of America and Germany
- Mario Arioli, Università LUM, Italy
- Selin Aslan, Argonne National Laboratory, United States of America
- Haim Avron, Tel Aviv University, Israel
- Zhaojun Bai, University of California, Davis, United States of America
- Jesse Barlow, The Pennsylvania State University, United States of America
- Christopher Beattie, Virginia Tech, United States of America
- Erna Begović Kovač, University of Zagreb, Croatia
- Peter Benner, Max Planck Institute for Dynamics of Complex Technical Systems, Germany
- Austin Benson, Cornell University, United States of America
- Michele Benzi, Scuola Normale Superiore, Pisa, Italy
- Luca Bergamaschi, University of Padua, Italy
- Paolo Bientinesi, Umeå University, Sweden
- Jan Blechta, Chemnitz University of Technology, Faculty of Mathematics, Germany
- Daniel Boley, University of Minnesota, USA
- Matthias Bolten, Bergische Universität Wuppertal, Germany
- Erik Boman, Sandia National Labs, USA
- Johannes Brust, Argonne National Laboratory, USA
- Alessandro Buccini, Università of Cagliari, Italy
- Aydin Buluç, Lawrence Berkeley Lab and UC Berkeley, United States of America
- Chao Chen, The University of Texas at Austin, United States of America
- Tyler Chen, University of Washington, USA
- Matthias Chung, Virginia Tech, United States of America
- Anna Concas, Department of Mathematics and Computer Science, University of Cagliari, Italy
- Alice Cortinovis, École Polytechnique Fédérale de Lausanne, Switzerland
• Anil Damle, Cornell University, United States of America
• Bart De Moor, ESAT-STADIUS KU Leuven, Belgium
• Eric de Sturler, Virginia Tech, USA
• Inderjit S. Dhillon, UT Austin and Amazon, United States of America
• Adii Dmytryshyn, Örebro University / Umeå University, Sweden
• Sergey Dolgov, University of Bath, United Kingdom
• Froilán Dopico, Universidad Carlos III de Madrid, Spain
• Zlatko Drmač, University of Zagreb, Croatia
• Vladimir Druskin, WPI, United States of America
• Fabio Durastante, Istituto per le Applicazioni del Calcolo ”M. Picone” - CNR, Italy
• Alan Edelman, MIT, United States of America
• Lars Eldèn Linköping University, Sverige
• Howard C. Elman, University of Maryland, United States of America
• Mark Embree, Virginia Tech, United States of America
• Massimiliano Fasi, Department of Mathematics, The University of Manchester, United Kingdom
• Dario Fasino, University of Udine, Italia
• Melina Freitag, Universität Potsdam, Germany
• Andreas Frommer, Bergische Universität Wuppertal Germany
• Isabella Furci, Wuppertal University, Germany
• Ion Victor Gosea, Max Planck Institute Magdeburg, Germany
• Pawan Goyal, Max Planck Institute, Magdeburg, Germany
• Anne Greenbaum, University of Washington, United States of America
• Chen Greif, The University of British Columbia, Canada
• Laura Grigori, Inria, Paris, France
• Luka Grubišić, University of Zagreb, Faculty of Science, Department of Mathematics, Croatia
• Nicola Guglielmi, Gran Sasso Science Institute, Italy
• Eric Hallman, North Carolina State University, United States of America
• Behnam Hashemi, Shiraz University of Technology, Iran
• Ken Hayami, National Institute of Informatics, Japan
• Reinhard Nabben, TU Berlin, Germany
• James Nagy, Emory University, United States of America
• Yuji Nakatsukasa, University of Oxford, United Kingdom
• Gian Maria Negri Porzio, School of Mathematics, The University of Manchester, United Kingdom
• Elizabeth Newman, Emory University, United States of America
• Esmond Ng, Lawrence Berkeley National Laboratory, United States of America
• Michael Kwok-Po Ng, University of Hong Kong, China
• Nancy K, Nichols, University of Reading, United Kingdom
• Vanni Noferini, Aalto University Finland
• Christopher Paige McGill University Canada
• Davide Palitta Max Planck Institute for Dynamics of Complex Technical Systems Germany
• Jan Papež, Institute of Mathematics, Czech Academy of Sciences, Czech Republic
• John Pearson, University of Edinburgh, United Kingdom
• Jennifer Pestana, University of Strathclyde, United Kingdom
• Martin Plešinger, Technical University of Liberec, Czech Republic
• Bor Plestenjak, IMFM and Faculty of Mathematics and Physics, University of Ljubljana, Slovenia
• Igor Pontes Duff, Max Planck Institute for Dynamics of Complex Technical Systems, Germany
• Yang Qi, INRIA Saclay and CMAP-Ecole Polytechnique, France
• María del Carmen Quintana Ponce, Universidad Carlos III de Madrid, Spain
• Lothar Reichel, Kent State University, USA
• Rosemary Renaut, Arizona State University, United States of America
• Emil Ringh, KTH Royal Institute of Technology, Sweden
• Leonardo Robol, Università di Pisa, Italy
• Jose Israel Rodriguez, UW Madison, USA
• Daniel Ruiz, University of Toulouse - IRIT, FRANCE
• Malena Sabaté Landman, Department of Mathematical Sciences, University of Bath, United Kingdom
• Michael Alan Saunders, Stanford University, United States of America
• Carmela Scalone, University of L’Aquila, Italy
• Meiyue Shao, Fudan University, China
• Punit Sharma, Indian Institute of Technology Delhi, India
• Edgar Solomonik, University of Illinois at Urbana–Champaign, United States of America
• Kirk M. Soodhalter, Trinity College, University of Dublin, Ireland
• Zdeněk Strakoš, Charles University, Prague, Czech Republic
• Brian Sutton, Randolph-Macon College, United States of America
• Daniel B. Szyld, Temple University, United States of America
• Jemima Tabeart, University of Edinburgh, United Kingdom
• Petr Tichý, Charles University, Prague, Czech Republic
• Françoise Tisseur, University of Manchester, United Kingdom
• Miroslav Tůma Charles University, Faculty of Mathematics and Physics, Czech Republic
• Roel Van Beeumen, Lawrence Berkeley National Laboratory, United States of America
• Paul Van Dooren, Universite catholique de Louvain, Belgium
• Bart Vandereycken, University of Geneva, Switzerland
• Simon Vary Mathematical Institute, University of Oxford United Kingdom
• Stephen Vavasis, University of Waterloo, Canada
• Nico Vervliet, KU Leuven, Belgium
• Matthias Voigt, University of Hamburg/TU Berlin, Germany
• Andy Wathen, Oxford University, United Kingdom
• David Watkins, Washington State University, United States of America
• Jonathan Weare, Courant Institute, NYU, United States of America
• Michal Wojtylak, Jagiellonian University, Krakow, Poland
• Yuanzhe Xi, Emory University, USA
• Xin Xing, Georgia Institute of Technology, United States of America
• Leihong Zhang, Soochow University, China
• Jörn Zimmerling, University of Michigan, United States of America
• Ian Zwaan, University of Wuppertal, Germany
A framework for second-order eigenvector centralities and clustering coefficients

Francesca Arrigo, Desmond J. Higham and Francesco Tudisco

Department of Mathematics and Statistics, University of Strathclyde
https://arrigofrancesca.wixsite.com/farrigo

Abstract

The most widely used approach to analyse complex systems in network science is to focus on pairwise interactions, that is, by studying adjacencies of nodes via edges. This is evident when we think about how networks are commonly represented. Indeed, if we let $G = (V, E)$ be an unweighted, undirected and connected network with $n$ nodes, then the most popular way of representing $G$ is via its adjacency matrix $A \in \mathbb{R}^{n \times n}$. The adjacency matrix is an extremely simple yet powerful tool: it encodes in a very efficient way the topology of the graph ($A_{ij} = 1$ if and only if $\{i, j\} \in E$) and it can also be easily manipulated using linear algebraic techniques. For this reason, this network representation is at the heart of several centrality and clustering algorithms. For example, when trying to identify the most important nodes within a graph, a popular approach is to rely on the ranking induced by a centrality measure and thus to select as the most important those entities that rank the highest. The majority of the most widely used centrality measures available in the literature are defined in terms of $A$ [8]. We focus here on the spectral measure called eigenvector centrality, which is defined for a node $i$ as the $i$th entry of the Perron eigenvector $y$ of $A$ [5, 12]:

$$y_i \propto \sum_{j=1}^{n} A_{ij} y_j.$$ 

This centrality measure is mutually-reinforcing, in the sense that it does not simply take into account the first-order interactions of node $i$ but also their quality: The importance score of node $i$ is proportional to the scores of its neighbours. The feature of mutual-reinforcement in a centrality measure allows to capture more information about the graph than what would be available using standard local approaches, where usually only the number of connections of node $i$ is taken into account (e.g., degree centrality).

In recent years, however, it has become apparent that just looking at first-order interactions is not a successful and comprehensive approach in network analysis. It has been observed that many important features of graphs arise from the interaction of larger groups of nodes [2, 10, 11]. This is a common phenomenon in social networks, for example, where people often interact in groups and not just in pairs. In their simplest form, these higher-order interactions occur among triplets of nodes instead of pairs. In fact, complex networks are well-known for containing a larger number of triangles than what would be expected in random graphs. This abundance of triangles was first observed in social interactions and it is usually associated with the idea of triadic closure, the mechanism for which two people having a common friend eventually become friends [6, 7]. The local density of triangles in a network is widely acknowledged as a meaningful network statistics and can be inferred from $A$. It is usually referred to as the Watts-Strogatz clustering coefficient and it is defined for each node as the ratio of closed triangles each node participates to the total number of triangles it could participate in [13]:

$$c_i = \frac{(A^3)_{ii}}{6(A^2)_{ii}((A^2)_{ii} - 1)}.$$
Information on triadic and higher-order interactions among nodes is indirectly used in many algorithms in network science by considering traversals around the network. However, recent work [3, 4, 7, 10] has shown that there is benefit in directly taking these into account when designing algorithms.

In this talk, we discuss a general tensor-based framework for incorporating second-order features, i.e., triangles, into network measures [1]. This approach allows us to combine traditional pairwise links with information that records whether triples of nodes are involved in wedges\(^1\) or triangles.

The underlying object of study is a constrained nonlinear eigenvalue problem associated with a cubic tensor \(T\):

\[ \alpha Ax + (1 - \alpha)T_p(x) = \lambda x, \]

where \(\alpha \in [0, 1]\) and

\[ T_p(x)_i = \sum_{j,k=1}^n T_{ijk} \left( \frac{|x_j|^p + |x_k|^p}{2} \right)^{1/p}. \]

The proposed framework includes as special cases eigenvector centrality \((\alpha = 1)\) and other recently proposed centrality measures from the literature [9]. It also allows for meaningful extensions, such as a mutually-reinforcing (spectral) version of the Watts-Strogatz clustering coefficient \((\alpha = 0)\).

Using recent results from nonlinear Perron–Frobenius theory [9], we establish existence and uniqueness of the solution \(x\) to the above eigenproblem under mild conditions on the network topology, and show that the new spectral measures can be computed efficiently using a nonlinear power method. We also give computational results on centrality and link prediction for real-world networks.

References


\(^1\)We recall that a wedge centred at node \(i\) is a path of length two of the form \(j - i - k\).


Solving matrix equations associated with random walks in the quarter plane: theoretical and algorithmic analysis

Dario A. Bini, Stefano Massei, Beatrice Meini, Jie Meng, Leonardo Robol

Dipartimento di Matematica, Università di Pisa
people.dm.unipi.it/bini

Abstract

We are interested in the numerical solution of unilateral quadratic matrix equations of the kind

\[ A_1 X^2 + A_0 X + A_{-1} = X \]  

where \( X \) is the matrix unknown and \( A_{-1}, A_0, A_1 \) are given semi-infinite matrices having nonnegative entries such that \( A = A_{-1} + A_0 + A_1 \) is irreducible and stochastic, i.e., \( A \mathbf{1} = \mathbf{1}, \mathbf{1} = (1, 1, \ldots)^T \).

These equations are encountered in the analysis of Quasi–Birth-Death (QBD) processes which describe a wide variety of queuing models [6], most of them are in the framework of bi-dimensional random walks. For a random walk in the quarter plane, the matrix coefficients \( A_i \) are semi-infinite and have the form

\[
A_i = \begin{bmatrix}
b_{i,0} & b_{i,1} & \cdots \\
a_{i,-1} & a_{i,0} & a_{i,1} & \cdots \\
a_{i,-1} & a_{i,0} & a_{i,1} & \cdots \\
& \ddots & \ddots & \ddots
\end{bmatrix}, \quad i = -1, 0, 1,
\]

where \( a_{i,j} \) are the probabilities that the particle involved in the random walk moves from a position of integer coordinates \((p, q), p, q > 0\) to a position of coordinates \((p+j, q+i)\) for \( i, j = -1, 0, 1, \) while \( b_{i,j} \) are the probabilities that the particle moves from \((0, q)\) to \((j, q+i)\) for \( j = 0, 1, i = -1, 0, 1.\)

This class of problems covers a wide variety of queuing models encountered in many areas of applied probability. The main demand from applications is to provide reliable methods for computing the invariant probability measure \( \pi = (\pi_{i,j}) \), i.e., the probabilities \( \pi_{i,j} \) that the particle is in \((i, j)\) when time tends to infinity.

The matrix analytic methodology (MAM) of M. Neuts [8], [6], provides an explicit representation of this invariant probability measure \( \pi \) in terms of the minimal nonnegative solution \( G \) of equation (1). This solution, which always exists, has nonnegative entries and is such that any other nonnegative solutions, if they exist, are larger than \( G \) entry-wise. This representation is the most effective and the most used tool for numerically computing \( \pi_{i,j} \) when the matrix coefficients \( A_i \) are finite matrices. As pointed out by Miyazawa in [7], in the case of semi-infinite matrices \( A_i \), despite the solution \( G \) still exists unique, the nice numerical features of the matrix analytic methodology are lost and one has to find alternative approaches to represent and to compute \( \pi_{i,j} \). On the other hand, the available approaches, valid in the infinite case, have substantially strong restrictions for their applicability [7].

In this talk we overcome the drawback pointed out by Miyazawa by extending and generalizing the recent results of [1], [2], [4]. In particular we extend the class \( QT \) of Quasi-Toeplitz matrices introduced in [1] by relying on the infinity norm instead of the 2-norm, and prove that the new class is a Banach algebra. Conditions are given in order that \( G \) belongs to \( QT \). In order to solve (1) we introduce three fixed point iterations and provide mild conditions under which the generated sequences converge to \( G \) in the infinity norm. We compare the speed of convergence of the three
iterations and provide explicit bounds to the rate of convergence when the initial approximation is the null matrix or a stochastic matrix and prove that with the latter choice the convergence speed is higher. In order to cover models from applications which include a reset of the system and which require a substantial modification of the random walk settings, we generalize the proposed matrix algebra by introducing the class \( \mathcal{EQT} \) of extended quasi-Toeplitz matrices, prove that it is a Banach algebra, and provide conditions under which the solution \( G \) belongs to \( \mathcal{EQT} \). The Matlab Toolbox CQT, introduced in [4] to operate with semi-infinite matrices, is modified accordingly to operate with matrices in \( \mathcal{QT} \) and \( \mathcal{EQT} \). Here are more formal details in this regard. For the full description of these results we refer to [5], [3].

Let \( W = \{ f(z) = \sum_{i \in \mathbb{Z}} f_i z^i : \sum_{i \in \mathbb{Z}} |f_i| < \infty \} \) be the Wiener algebra and associate with \( f(z) \in W \) the semi-infinite Toeplitz matrix \( T(f) = (f_{j-i})_{i,j \in \mathbb{Z}^+}. \) Let \( \ell^\infty \) be the set of sequences \( x = (x_i)_{i \in \mathbb{Z}^+} \) such that \( ||x||_\infty = \sup_{i} |x_i| < +\infty \) and let \( \mathcal{L}^\infty \) denote the set of semi-infinite matrices \( A \) having bounded infinity norm, i.e., \( ||A||_\infty = \sup_{x \in \ell^\infty \setminus \{0\}} \|Ax\|_\infty / \|x\|_\infty < +\infty. \) Denote \( \mathcal{D} \subset \mathcal{L}^\infty \) the class of matrices \( D = (d_{i,j}) \) having the decay property, i.e., \( \lim_i \sum_j |d_{i,j}| = 0. \) We define the class of quasi Toeplitz matrices \( \mathcal{QT} = \{ T(a) + E : a \in W, E \in \mathcal{D} \} \) and the class of extended quasi-Toeplitz matrices \( \mathcal{EQT} = \{ A = B + 1v^T : B \in \mathcal{QT}, v \in \ell^\infty \}. \) We prove that \( \mathcal{QT} \) and \( \mathcal{EQT} \), endowed with the infinity norm, are Banach algebras. Moreover, we prove that the minimal nonnegative solution of (1) where the coefficients \( A_t \) have the form (2), can be written as \( G = T(g) + E_g, \) where \( E_g \in \mathcal{L}^\infty, \) \( g(z) = \sum_{i \in \mathbb{Z}} g_i z^i \in W \) and for any \( z \in \mathbb{C}, |z| = 1, \) \( g(z) \) is the solution of minimum modulus of the scalar equation

\[
a_1(z)g(z)^2 + a_0(z)g(z) + a_{-1}(z) = g(z).
\]

This fact provides an effective algorithm for approximating the coefficients \( g_i \) by relying on the FFT-based technique of evaluation/interpolation. We prove that, if \( \sum_{j=-1}^{1} a_{-1,j} > \sum_{j=-1}^{1} a_{1,j}, \) that is the overall probability that the random walk evolves downwards is higher than the overall probability that the random walk evolves upwards, then \( E_g \in \mathcal{D}, \) that is, \( G \in \mathcal{QT}. \) We also prove that, if this condition is not satisfied, but the Markov chain is strongly ergodic, then \( E_g - 1v^T \in \mathcal{D} \) for a suitable vector \( v \in \ell^\infty \) so that \( G \in \mathcal{EQT}. \) These properties allow to design algorithms for computing \( G \) based on the (modified) CQT-Toolbox of [4], relying on three fixed point iterations. These iterations are induced by the following functions \( F_1(X) = A_1 X^2 + A_0 X + A_{-1}, \) \( F_2 (X) = (I - A_0)^{-1} (A_1 X^2 + A_{-1}), \) \( F_3(X) = (I - A_0 - A_1 X)^{-1} A_{-1}. \) Numerical experiments show that in many cases these fixed point iterations outperform quadratically convergent algorithms like Logarithmic Reduction of [6].

**References**


Some Distance Problems for Matrix Polynomials via Block Toeplitz Matrices

Shreemayee Bora, and Biswajit Das

Department of Mathematics, Indian Institute of Technology Guwahati
https://www.iitg.ac.in/shbora/

Abstract

Two important distance problems associated with an $n \times n$ matrix polynomial $P(\lambda) = \sum_{i=0}^{k} \lambda^i A_i$ of degree $k$ are considered. The first one is the distance to a nearest regular matrix polynomial with an elementary divisor associated with a finite or infinite eigenvalue $\lambda_0$ of specified minimal length, say, $r > 1$. The second one is the distance to a nearest rank deficient matrix polynomial. Both problems pose theoretically intriguing questions of very significant practical importance [3, 5, 2]. In fact finding a practical numerical method for computing the second distance is a long standing open question [2]. It is observed that certain block Toeplitz matrices play an important role in the computation of both the distances resulting in a strong connection between them.

To solve the first distance problem it is enough to find the distance to a nearest matrix polynomial having $\lambda_0$ as an eigenvalue of algebraic multiplicity at least $r$. Unlike most of the work in the literature, it is considered for a $P(\lambda)$ that is either regular or singular with perturbations affecting all the coefficient matrices. In fact by using elementary perturbation theory it is shown that if the matrix polynomial $P(\lambda)$ is singular, then it is arbitrarily close to a regular matrix polynomial with the desired property. This solves the distance problem for singular matrix polynomials. For regular polynomials, the distance problem is shown to be equivalent to finding a smallest structure preserving perturbation such that certain block Toeplitz matrices formed by $P(\lambda_0)$ and its derivatives becomes suitably rank deficient. From this it follows that if $\lambda_0$ is not an eigenvalue of $P(\lambda)$, then solving the distance problem is equivalent to computing a generalized version of a structured singular value or $\mu$-value [4]. It is well known that the $\mu$-value computation is an NP-hard problem [1]. Therefore this problem is also likely to be NP-hard. The distance is formulated as an optimization and bounds are derived from the characterizations. A special case for which the solution has a closed form expression is presented. Computed values of the distance obtained via BFGS and Matlab’s globalsearch algorithms are compared with upper and lower bounds which appear to be tight in many instances.

The second problem is addressed from the more general perspective of finding the distance from a regular matrix polynomial to a nearest matrix polynomial of rank at most $r$ for a specified positive integer $r (\leq n-1)$. This general problem is shown to be equivalent to computing a smallest structure preserving perturbation such that certain convolution matrices of the polynomial (that are block Toeplitz) become suitably rank deficient, thus leading to a characterization of the distance to singularity. Additionally by considering the relationship of the Jordan chain structure of a matrix polynomial at 0 with its rank, the distance to singularity is shown to be also equivalent to the rank deficiency of another type of block Toeplitz matrix that appears in the first distance problem when $\lambda_0 = 0$. This leads to a new characterization of the distance to singularity that links the two distance problems under consideration. The characterizations imply that finding the distance to singularity is an NP-hard problem as it is the reciprocal of a certain $\mu$-value. Upper and lower bounds as well as information about the minimal indices of nearest singular matrix polynomials are derived from these characterizations. Based on this information, a numerical strategy to compute the distance to singularity is devised and implemented via BFGS and Matlab’s globalsearch.
algorithms. Numerical experiments show that the bounds are tight in many cases and the computed distances compare favourably with values obtained in the literature, thus demonstrating the efficacy of the strategy.

References


Iterative Refinement in Three Precisions

Erin C. Carson, Nicholas J. Higham, and Srikara Pranesh

Faculty of Mathematics and Physics, Charles University, Prague

www.karlin.mff.cuni.cz/~carson

Abstract

Support for floating point arithmetic in multiple precisions is becoming increasingly common in emerging architectures. For example, half precision is now available in the NVIDIA P100 and V100 GPUs and the AMD Radeon Instinct MI25 GPU, on which it runs twice as fast as single precision with a proportional savings in energy consumption. It is anticipated that future exascale machines will support a variety of precisions ranging from quarter to quadruple. The ability to exploit such hardware for performance gains presents a huge opportunity in numerical linear algebra.

In this presentation, we take a first step in this direction. We present and analyze a general algorithm for solving an $n \times n$ nonsingular linear system $Ax = b$ based on iterative refinement in three precisions. Our results provide the ability to solve certain linear systems $Ax = b$ at up to twice the speed without any sacrifice to accuracy compared to traditional iterative refinement approaches. Recent work implementing our approach has shown such speedups to be attainable on multiprecision hardware in practice [4].

Iterative refinement is a method for improving an approximate solution $y$ to a linear system $Ax = b$ by computing the residual $r = b - Ay$, solving the correction equation $Ad = r$, forming the update $y \leftarrow y + d$, and repeating these steps as necessary. We consider a general iterative refinement algorithm that includes a variety of existing ones as special cases. The algorithm contains three precisions: $u$ is the working precision in which the data $A, b$ and the solution $x$ are stored, $u_f$ is the precision at which a factorization of $A$ is computed, and $u_r$ is the precision at which the residuals are computed, where it is assumed that $u_r \leq u \leq u_f$. Our analysis also relies on a fourth “effective precision” $u_s$, which is the precision at which the correction equation is effectively solved; $u_s$ will either take on the value $u$ or $u_f$ in cases of interest.

Our results generalize and unify many existing rounding error analyses for iterative refinement, including traditional iterative refinement in which $u_f = u$ and $u_r = u^2$ (see, e.g., the analyses of Wilkinson [10] and Moler [7]), fixed precision refinement where $u_f = u = u_r$ (see Janowski and Woźniakowski [5] and Skeel [9]), and the more recent low-precision factorization variant (e.g., Langou et al. [6]), with $u_f$ single precision and $u = u_r$ double precision. Additionally, our analysis allows for a new class of iterative refinement approaches where all three precisions differ. For example, we can take $u_f$ half precision, $u$ single precision, and $u_r$ double precision. We could even go further and take $u_f$ half precision, $u$ double precision, and $u_r$ quad precision. Using three different precisions allows the best of both worlds; we obtain the performance benefits of the low precision factorization combined with the accuracy of traditional iterative refinement.

We provide a complete rounding error analysis and derive sufficient conditions for convergence and bounds for the attainable forward error and normwise and componentwise backward errors; see [3]. For example, with single precision as the working precision, using LU factorization in IEEE half precision as the solver and calculating the residuals in double precision it is possible to solve $Ax = b$ to full single precision accuracy for $\infty$-norm condition numbers $\kappa_\infty(A) \leq 10^4$, with the $O(n^3)$ part of the computations carried out entirely in half precision.

Our analysis contains two main innovations. First, we allow for a general solver for the correction equation (which may be a different solver than that used to compute the initial approximate
solution. We require only that the solver satisfy three assumptions. First, the normwise relative error for the correction solve in each refinement step must be bounded by a multiple of $u_\text{s}$ and must be less than 1. The second and third assumptions are related to the normwise relative backward error and componentwise relative backward error, which much both be bounded in terms of $O(u_\text{s})$.

Our second innovation allows us to obtain tighter upper bounds. Instead of the commonly-employed bound $\|A(x - x_i)\| \leq \|A\|\|x - x_i\|$, which can be quite pessimistic in practice, we instead define the quantity $\mu_i$ to be such that $\|A(x - x_i)\| = \mu_i\|A\|\|x - x_i\|$. By reasoning about the size of the components of the residual in the left singular vectors of $A$, we can draw conclusions about how $\mu_i$ changes over the course of the refinement. In short, we expect that $\mu_i$ begins close to $\kappa(A)^{-1}$ and grows to its upper bound of 1 only once the forward error is very small.

We show further that by solving the correction equations by GMRES preconditioned by the LU factors the restriction on the condition number can be weakened to $\kappa_\infty(A) \leq u_f^{-1}u^{-1/2}$, allowing the 3-precision approach to apply to more ill conditioned systems. We call this approach GMRES-IR.

We prove that if the application of the preconditioned operator is performed in precision $u^2$, then the GMRES backward stability results of Paige, Rozložník, and Strakoš [8] hold for the preconditioned system; see [2]. This result is then used to prove that the update equation is solved with some degree of relative accuracy, even in the case that $A$ is extremely ill conditioned. We further discuss the convergence behavior of the resulting preconditioned GMRES method and complementary techniques which can be used to further accelerate convergence such as Krylov subspace recycling.

We will briefly discuss recent work involving the extension of our approach to least squares problems. Using the augmented matrix formulation of Björck [1], the iterative refinement approach for least squares is the same as for linear systems, and thus the theoretical analysis immediately applies. However, in the least squares case, we would like to use an existing QR factorization of $A$ rather than explicitly forming the augmented matrix and computing its LU factorization. We show that QR factors of $A$ can be used to construct various preconditioners (some satisfying the theory and others that work well in practice) such that a GMRES-IR-like technique can be employed.

We conclude with a broader perspective regarding the cost of iterative computations and the challenge of evaluating tradeoffs between performance and accuracy on modern hardware.

References


Asynchronous Iterative Methods

Edmond Chow

School of Computational Science and Engineering, Georgia Institute of Technology
www.cc.gatech.edu/~echow

Abstract

A parallel asynchronous iterative method for solving a system of equations is one in which processors do not synchronize at the end of each iteration. Instead, processors proceed iterating on the equations assigned to it with the latest data that is available from other processors. Running an iterative method in such an asynchronous fashion may reduce solution time when there is an imbalance of the effective load between the processors. Solution time may also be reduced when communication costs are high because computation continues while communication takes place.

Consider the \( n \times n \) system of equations \( x = G(x) \), which can be written in scalar form as \( x_i = g_i(x) \), \( i = 1, \ldots, n \). An asynchronous iterative method for solving this system of equations can be defined mathematically as the sequence of updates:

\[
x_i^{(j)} = \begin{cases} 
    x_i^{(j-1)}, & \text{if } i \notin J_j \\
    g_i(x_{s_1^i(j)}^{(j-1)}, x_{s_2^i(j)}^{(j-1)}, \ldots, x_{s_n^i(j)}^{(j-1)}), & \text{if } i \in J_j 
\end{cases}
\]

where \( x_i^{(j)} \) denotes \( x_i \) at time instant \( j \), \( J_j \) is the set of indices of variables being updated at instant \( j \), and \( s_k^i(j) \) is the instant that \( x_k \) is read when computing \( g_i \) at instant \( j \). In contrast to standard, synchronous iterative methods, (1) not all updates are performed at the same time instant; (2) updates may use stale information, which models communication delays in reading or writing the variables.

With some natural assumptions on the sequence of updates above, much work has been done on showing the conditions under which asynchronous iterative methods converge (see the survey [4]). For linear systems, asynchronous iterations converge for any initial guess if and only if \( \rho(|T|) < 1 \), where \( T \) is the iteration matrix for the standard, synchronous iterations, and \( |\cdot| \) is taken entrywise. Since \( \rho(T) \leq \rho(|T|) \), it appears that the conditions for convergence of asynchronous iterations is more strict than those for convergence of synchronous iterations.

In our work, we address the practical convergence behavior of asynchronous iterative methods as opposed to asymptotic convergence. Note that asynchronous iterations differ from synchronous iterations in convergence behavior (from the use of stale values, for example) and in computational efficiency (from removing synchronizations and data copies, for example). In order to study convergence behavior independently of computational efficiency, our approach is to simulate asynchronous iterations, i.e., execute the mathematical model with parameters that control (1) the frequency at which updates are performed, and (2) the staleness of the data that is used. In particular, we simulate asynchronous iterations with an “update probability” that is the probability that a component of \( x \) is updated at a given time instant, and a “delay bound” which is a bound on the number of time instants in the past from which data might be read.

Through this approach, we show the surprising result that the asynchronous Jacobi method can converge when the corresponding synchronous method does not converge [7, 8]. This does not contradict the theory since, although there exists a sequence of asynchronous iterations that does not converge when \( \rho(|T|) > 1 \), sequences that do converge can be more common. We will shed light on
the practical convergence behavior of asynchronous iterative methods by showing their similarities to synchronous multiplicative iterations (such as Gauss-Seidel) and weighted Jacobi methods. We also compare the results of simulating the mathematical model with actual asynchronous computations in order to verify the accuracy of the model.

We further consider asynchronous versions of several other iterative methods, including optimized Schwarz, the Chebyshev semi-iterative method, and multigrid.

The optimized Schwarz method can be a rapidly converging fixed-point iteration. Simulations and actual implementations of asynchronous optimized Schwarz show that its convergence behavior and solution time can be better than for synchronous optimized Schwarz [10, 5].

Chebyshev iterations are not fixed-point iterations and thus the existing theory for asynchronous methods does not apply. However, theory for the inexact Chebyshev method [6] suggests that asynchronous Chebyshev may converge under appropriate conditions, e.g., when the spectrum of the iteration matrix of the method being accelerated is, or is estimated to be, narrow.

We also define asynchronous versions of multigrid methods. Here, we desire that coarse grid corrections from each level of the multigrid hierarchy can perform updates to the fine grid solution asynchronously. Thus we are interested in asynchronous versions of additive multigrid methods that can be used as solvers rather than just as preconditioners. With parallel implementations, we are able to show that asynchronous versions of additive multigrid methods can retain grid-independent convergence behavior and reduce solution time compared to synchronous multiplicative multigrid when communication costs are high or when a large number of processors are used [9].

References


Efficient Krylov Subspace Methods for Uncertainty Quantification in Large Linear Inverse Problems

Julianne Chung, Katrina Petroske, and Arvind K. Saibaba

Department of Mathematics, Virginia Tech
http://intranet.math.vt.edu/people/jmchung/

Abstract

Inverse problems arise in various scientific applications, and a significant amount of effort has focused on developing efficient and robust methods to compute approximate solutions. However, as these numerical solutions are increasingly being used for data analysis and to aid in decision-making, there is a critical need to be able to obtain valuable uncertainty information (e.g., solution variances, samples, and credible intervals) to assess the reliability of computed solutions. Tools for inverse uncertainty quantification (UQ) often build upon the Bayesian framework from statistical inverse problems. However, for problems with a very large number of unknown parameters (e.g., dynamic inverse problems) and for problems where computation of the square root and inverse of the prior covariance matrix are prohibitively expensive, UQ for linear inverse problems remains a challenging task.

In this talk, we describe efficient Krylov subspace iterative methods, previously used for solving weighted least-squares problems, for large-scale UQ in inverse problems. For concreteness, we consider linear inverse problems of the form

\[ d = As + \delta, \]

where the goal is to reconstruct the desired parameters \( s \in \mathbb{R}^n \), given forward operator (or parameter-to-observable map) \( A \in \mathbb{R}^{m \times n} \) and the observed data \( d \in \mathbb{R}^m \). The inverse problem involves estimating the unknown parameters \( s \) from the data \( d \), and the recovery of these parameters is typically an ill-posed problem. We adopt a Bayesian approach where we assume that the measurement errors \( \delta \) and the unknowns \( s \) are mutually independent Gaussian variables, i.e., \( \delta \sim \mathcal{N}(0, R) \) and \( s \sim \mathcal{N}(\mu, \lambda^{-2}Q) \) where \( R \) and \( Q \) are symmetric positive definite matrices, \( \mu \in \mathbb{R}^n \), and \( \lambda \) is a scaling parameter that controls the prior precision and is also known as the regularization parameter. Using Bayes’ theorem and under our assumptions, the posterior distribution has the following probability density function,

\[
\pi_{\text{post}} \propto \exp \left( -\frac{1}{2} \| As - d \|_R^{-1} - \frac{\lambda^2}{2} \| s - \mu \|_Q^{-1} \right),
\]

where \( \| x \|_M = \sqrt{x^\top M x} \) is a vector norm for any symmetric positive definite matrix \( M \) and \( \propto \) means “proportional to.” Thus, the posterior distribution is Gaussian, with corresponding measure \( \rho_{\text{post}} = \mathcal{N}(s_{\text{post}}, \Gamma_{\text{post}}) \), where the posterior covariance and mean are given as

\[
\Gamma_{\text{post}} = (\lambda^2 Q^{-1} + A^\top R^{-1} A)^{-1} \quad \text{and} \quad s_{\text{post}} = \Gamma_{\text{post}}(A^\top R^{-1} d + \lambda^2 Q^{-1} \mu)
\]

respectively [2]. In the Bayesian framework, the solution to the inverse problem is the posterior distribution. However, for practical interpretation and data analysis, it is necessary to describe various characteristics of the posterior distribution [1, 9].

We exploit various tools from numerical linear algebra to efficiently explore the posterior.
• For the problems of interest, computing the inverse and square root of $R$ are inexpensive (e.g., $R$ is a diagonal matrix), but explicit computation of $Q$ (or its inverse or square root) may not be possible. We assume that matrix-vector multiplications with $A$, $A^\top$, and $Q$ can be done efficiently as function evaluations, and we use generalized Golub-Kahan based iterative methods [4] to compute an estimate of $s_{\text{post}}$. The regularization parameter $\lambda$ and stopping iteration can be selected simultaneously and automatically.

• We use elements from the generalized Golub-Kahan bidiagonalization to approximate the posterior covariance matrix $\Gamma_{\text{post}}$, and we provide theoretical results that quantify the accuracy of the approximate posterior covariance matrix and of the resulting posterior distribution.

• We describe methods that use the approximate posterior distribution to compute measures of uncertainty.

• We present two methods that use preconditioned Lanczos methods to efficiently generate samples from the posterior distribution. Previous work on Lanczos methods for sampling from Gaussian distributions can be found in, e.g., [6, 8, 3], but these algorithms are meant for sampling from generic Gaussian distributions and do not exploit the structure of the posterior covariance matrix as we do.

To demonstrate the effectiveness and efficiency of the described approaches, we provide numerical examples from dynamic tomography applications, where the goal is to reconstruct a sequence of images from a sequence of projection datasets. Such scenarios are common in dynamic photoacoustic or dynamic electrical impedance tomography, where the underlying parameters change during the data acquisition process [5].

Additional details for this work can be found in [7].

References


Abstract

In statistics and machine learning, ridge regression (also known as Tikhonov regularization or weight decay) is a variant of regularized least squares problems where the choice of the penalty function is the squared $\ell_2$-norm. Formally, let $A \in \mathbb{R}^{n \times d}$ be the design matrix and let $b \in \mathbb{R}^n$ be the response vector. Then, the linear algebraic formulation of the ridge regression problem is as follows:

$$x^* = \arg\min_{x \in \mathbb{R}^d} \left\{ \|Ax - b\|_2^2 + \lambda\|x\|_2^2 \right\},$$

where $\lambda > 0$ is the regularization parameter. There are two fundamental motivations underlying the use of ridge regression. First, when $d \gg n$, i.e., the number of predictor variables $d$ greatly exceeds the number of observations $n$, fitting the full model without regularization (i.e., setting $\lambda$ to zero) will result in large prediction intervals and a non-unique regression estimator. Second, if the design matrix $A$ is ill-conditioned, solving the standard least-squares problem without regularization would depend on $(A^TA)^{-1}$. This inversion would be problematic if $A^TA$ were singular or nearly singular and thus adding even a little noise to the elements of $A$ could result in large changes in $(A^TA)^{-1}$. Due to these two considerations, solving standard least-squares problems without regularization may provide a good fit to the training data but may not generalize well to test data.

Ridge regression abandons the requirement of an unbiased estimator in order to address the aforementioned problems. The minimizer of eqn. (1) is $x^* = A^T(\lambda A^TA + \lambda I_n)^{-1}b$. It is easy to see that $x^*$ can be computed in time $O(nd \min\{n,d\})$. In our talk, we will focus on design matrices $A \in \mathbb{R}^{n \times d}$ with $d \gg n$, which is the most common setting for ridge regression. The recent flurry of activity on Randomized Linear Algebra (RLA) and the widespread use of sketching as a tool for matrix computations, resulted in many novel results for ridge regression; see [1] for a detailed discussion of prior work.

We will present a novel iterative algorithm for sketched ridge regression and two simple sketching-based structural conditions under which the proposed algorithm guarantees highly accurate approximations to the optimal solution $x^*$. More precisely, our approach guarantees that, as long as a simple structural constraint is satisfied, the resulting approximate solution vector $\hat{x}^*$ satisfies (after $t$ iterations)

$$\|x^* - \hat{x}^*\|_2 \leq \varepsilon^t\|x^*\|_2.$$

Error guarantees of the above form are highly desirable. Indeed, beyond being a relative error guarantee, the dependency on $\varepsilon$ drops exponentially fast as the number of iterations increases. It is easy to see that by setting $\varepsilon^t = \varepsilon^t$, $O(\ln(1/\varepsilon^t))$ iterations would suffice to provide a relative error guarantee with accuracy parameter $\varepsilon^t$. This means that converging to, say, ten decimal digits of accuracy would necessitate only a constant number of iterations. (See [1] for a comparison of this bound with prior work.)

Let $V \in \mathbb{R}^{d \times n}$ be the matrix of right singular vectors of $A$ (assume, for simplicity of presentation, that $A$ has rank $n$). For eqn. (2) to hold, a sketching matrix $S \in \mathbb{R}^{d \times s}$ is to be constructed such
that (for an appropriate choice of the sketching dimension $s \ll d$)

$$
\|V^T S S^T V - I_n\|_2 \leq \frac{\varepsilon}{2}.
$$

(3)

We note that the constraint of eqn. (3) has been the topic of intense research in the Randomized Linear Algebra literature; this is precisely the reason why we use eqn. (3) as the building block in our analysis. A variety of oblivious sketching matrix constructions for $S$ can be used to satisfy eqn. (3).

However, one deficiency of the structural constraint of eqn. (3) is that all known constructions for $S$ that satisfy the constraint need a number of columns $s$ that is proportional to $n$. As a result, the running time of any algorithm that computes the sketch $AS$ is also proportional to $n$ and the running time of the overall algorithm is $O(t \cdot \text{nnz}(A)) + \tilde{O}(n^3/\varepsilon^2)$. (The $\tilde{O}$ hides poly-logarithmic factors.)

In the context of ridge regression, an important quantity is the effective degrees of freedom:

$$
d_\lambda = \sum_{i=1}^n \sigma_i^2 / (\sigma_i^2 + \lambda) \leq n,
$$

where $\sigma_i$ are the singular values of $A$. It would be much better to design algorithms whose running time depends on the degrees of freedom $d_\lambda$, which is upper bounded by $n$, but could be significantly smaller depending on the distribution of the singular values and the choice of $\lambda$. We propose a second structural constraint that reduces the number of columns $s$ of the matrix $S$ to depend on $d_\lambda$, which could be considerably smaller than $n$. However, this improved dependency on $d_\lambda$ instead of $n$ comes with a mild loss in accuracy, resulting in an additive-relative error guarantee, while reducing the running time to $O(t \cdot \text{nnz}(A)) + \tilde{O}(d_\lambda n^2/\varepsilon^2)$. (See [1] for details.)

Our algorithm can also be viewed as a preconditioned Richardson iteration for solving the linear system $(A A^T + \lambda I_n)y = b$ with pre-conditioner $P^{-1} = (S S^T A + \lambda I_n)^{-1}$ and step-size equal to one.

Our analysis offers several advantages over preconditioned Richardson iteration: importantly, in our case, $P^{-1}(A A^T + \lambda I_n)$ is not symmetric positive definite which, according to existing literature, implies that the convergence of Richardson’s method is monotone in terms of the energy-norm induced by $A A^T + \lambda I_n$, but not the Euclidean norm.

We will also discuss applications of fast iterative algorithms for ridge regression to fundamental problems such as Fischer Discriminant Analysis (FDA) and the analysis of Inexact Interior Point Methods (IIPMs) to approximately solve linear programs. With respect to FDA, we will present a simple, fast, randomized, iterative, sketching-based algorithm that leverages the sketching-based framework for ridge-regression and comes with provable accuracy guarantees when compared to the conventional approaches. We prove that accurate approximations can be achieved by using a sketching matrix whose size depends on the effective degrees of freedom of the RFDA problem [2].

With respect to IIPMs, we will present a novel analysis of inexact interior point methods when sketching-based approaches are used to solve the least-squares regression problem that arises when looking to update the Newton search direction. Our approach essentially uses a sketch of the input matrix as a pre-conditioner to solve the aforementioned problem and carefully analyzes the effect of obtaining an approximate solution at each iteration to the convergence rate and speed of the overall IIPM [2].

Finally, empirical evaluations supporting our theoretical analyses for all problems will also be presented.

References


Use of Partitioning in Solving Sparse Linear Equations on Heterogeneous High Performance Computers

Iain Duff

SCD, RAL, UK and Cerfacs, France
http://www.numerical.rl.ac.uk/people/isd/isd.html

Abstract

We discuss the solution of systems of linear equations

\[ Ax = b \]

where the matrix \( A \) is large and sparse. We use direct methods based on a matrix factorization of the form

\[ PAQ = LU \]

where \( P \) and \( Q \) are permutation matrices and \( L \) and \( U \) are lower and upper triangular matrices, respectively. When \( A \) is sparse it is important that the factors retain much of the original sparsity. Sparse direct methods are often justifiably the method of choice for solving sparse systems but their main limitation is that they can be costly in storage and operations and are not generally linear in problem size. Although the complexity of sparse factorizations are considerably better than the \( O(n^3) \) for dense factorizations for systems of size \( n \), the complexity is typically \( O(n^\alpha) \), where \( \alpha \) is between 1 and 2. For example, the complexity for factorizing a matrix of order \( n \) from a finite difference discretization of Poisson’s equation can be proved to have a value of \( \alpha \) of \( 3/2 \) and 2 for 2D and 3D problems, respectively.

We will discuss two classes of methods that exploit this nonlinearity through partitioning the system. If for example, we partitioned the system into \( k \) perfectly balanced pieces then we would get a reduction in complexity of \( k^\alpha - 1 \). Another major benefit of partitioning is that work on the subsystems is independent and we exploit that when running on heterogeneous parallel computers with node parallelism within subsystems and distributed parallelism from the partitioning.

Our first scheme, uses Zoltan[1] to order the matrix to singly bordered block diagonal form (SBBDF). We then use the multicore code ParSHUM[3] to factorize the blocks on the diagonal and assign an MPI process to each such factorisation. We illustrate this with some runs on power systems applications from DigSILENT GmbH in Table 1. Note that these speedups are in addition good single node parallelism.

We have compared our code with the state-of-the-art codes MUMPS and SuperLU and, on the matrices in Table 1, we are typically 4-5 times faster than MUMPS and up to 30 times faster than SuperLU.

Our second use of partitioning is in the context of a hybrid method based on the block Cimmino algorithm. Here the matrix is partitioned into strips and the corresponding equations for each strip are solved simultaneously by a direct method using an augmented system. The method calculates the next iterate by summing the solutions from the subsystems [4]. The convergence depends on the angles between these subspaces and we found it beneficial to replace our use of PaToH [2] by a numerically-aware partitioning algorithm that tries to make the subspaces orthogonal [5, 6].
Table 1: The execution time in seconds for ParSHUM on matrices partitioned in SBBDF.

<table>
<thead>
<tr>
<th>Matrix</th>
<th>n</th>
<th>nnz</th>
<th>#MPI processes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$10^3$</td>
<td>$10^3$</td>
<td>1</td>
</tr>
<tr>
<td>InnerLoop1</td>
<td>197</td>
<td>745</td>
<td>0.15</td>
</tr>
<tr>
<td>Jacobian_unbalancedLdf</td>
<td>203</td>
<td>2410</td>
<td>1.11</td>
</tr>
<tr>
<td>Newton_Iteration</td>
<td>427</td>
<td>2380</td>
<td>0.49</td>
</tr>
<tr>
<td>Newton_detailed</td>
<td>7355</td>
<td>23741</td>
<td>7.39</td>
</tr>
</tbody>
</table>

show speedups of our code in Figure 2. We see that the number of iterations expectedly increases when we have more partitions but this is more than compensated for by the reduction in direct factorization time, particularly so for the matrix Cage13 where superlinear behaviour is evident.

![Figure 2: Convergence of the method when the number of partitions increases from 8 to 64 running with one MPI process per partition on a single node. Numbers on top of each bar correspond to the number of CG iterations, using a threshold of $10^{-8}$.](image)

Although this talk is singly-authored, we will use illustrations from ongoing joint work with colleagues from the EU Horizon 2020 FET-HPC NLAFET Project (2015-2019) (Cayrols, Lopez, and Nakov) and in Toulouse at the Cerfacs-IRIT Common Lab (Leleux, Ruiz, and Torun).

References


Laboratory, 2019. NLAFET Working Note 22. Submitted to SIMAX.


Krylov Methods for Low-Rank Regularization

Silvia Gazzola, Chang Meng, and James G. Nagy

Department of Mathematical Sciences, University of Bath
https://people.bath.ac.uk/sg968/

Abstract

Regularization should be applied to recover a meaningful approximation of a solution \( x \in \mathbb{R}^N \) of a large-scale linear discrete inverse problem of the form

\[
Ax + n = b,
\]

(1)

where \( A \in \mathbb{R}^{M \times N} \) and \( n \) is some unknown Gaussian white noise. Problems like (1) arise in a variety of imaging applications, where \( x = \text{vec}(X) \) is a “vectorialized” version of a two-dimensional image \( X \in \mathbb{R}^{n \times n} \), with \( n = \sqrt{N} \) (i.e., \( x \) is obtained by stacking the columns of \( X \)). We focus on variational regularization methods that approximate \( x \) by a quantity \( x_{\text{reg}} \) computed as follows

\[
x_{\text{reg}} = \arg \min_{x \in C} \|Ax - b\|_2^2 + \lambda R(x),
\]

(2)

where \( C \) is a set of constraints, \( \lambda > 0 \) is a regularization parameter, and \( R(\cdot) \) is a regularization term. The success of the regularization method (2) crucially depends on the choice of \( C \) and \( R(\cdot) \), which should encode known information about \( x \), and the parameter \( \lambda \), which balances the least squares data-fitting term and the regularization term.

In this talk we will present new algorithms that enforce a low-rank solution \( x_{\text{reg}} = \text{vec}(X_{\text{reg}}) \) of (2), which is meaningful in imaging tasks such as image deblurring, computed tomography, and image inpainting.

The literature on low-rank solvers for (1) is already quite prolific. Here we distinguish between two main classes of available methods. On one side we have Krylov-like projection methods, which incorporate explicit rank truncation operations for the basis vectors of the solution subspace generated by an Arnoldi-like algorithm; referring to (2), this corresponds to choosing \( R(x) = 0 \) and \( C \) as the set of “vectorialized” matrices having a prescribed rank \( \rho \geq 1 \). These methods are quite heuristic (especially when it comes to the choice of the truncation rank \( \rho \)), and so far have only been employed to solve structured well-conditioned linear systems (see [3]). On the other side we have a plethora of optimization methods for nuclear norm regularization (NNR), which implicitly impose low-rank constraints by penalizing the magnitude of the singular values of the “vectorialized” matrix \( x_{\text{reg}} \); referring to (2), NNR corresponds to choosing \( C = \mathbb{R}^N \), and \( R(x) \) as the nuclear norm

\[
R(x) = \|X\|_{*,p} = \sum_{i=1}^{n} (\sigma_i(X))^p, \quad 0 < p \leq 1,
\]

(3)

where \( X = U_X \Sigma_X V_X^T \), \( \Sigma_X = \text{diag}(\sigma_1(X), \ldots, \sigma_n(X)) \), is the singular value decomposition (SVD) of \( X \). NNR is theoretically well-motivated. One common approach for solving the NNR problem consists in iteratively reweighting the regularization term (3), so that the original NNR problem is reformulated as a sequence of quadratic problems that are handled by gradient descent (see [4, 5]).

Our new algorithms can be placed at the interface of the two main classes of methods described above: in particular, by iteratively reweighting the nuclear norm regularization term (3), the new
methods are naturally associated to an inner-outer iteration scheme (where each quadratic problem in the sequence is solved via an inner cycle of iterations, and updated weights for the next quadratic problem in the sequence are computed at each outer iteration). The main elements of novelty of the proposed algorithms, with respect to available low-rank solvers, can be outlined as follows:

(a) Kronecker product properties are exploited to define an appropriate reweighted 2-norm regularization term for each quadratic problem in the sequence, the \((k + 1)\)th being

\[
x_{k+1} = \arg \min_x \|Ax - b\|_2^2 + \frac{\lambda p}{2} \left( \frac{1}{2} \left( I \otimes (\Sigma_{X_k}^2 + \gamma I)^\rho/4 - 1/2 \right) \right) \left( V_{X_k}^T \otimes U_{X_k}^T \right) x_2,
\]

where \(X_k = U_{X_k} \Sigma_{X_k} V_{X_k}^T\) is the SVD of the (approximate) solution \(x_k = \text{vec}(X_k)\) of the \(k\)th quadratic problem, \(k = 1, 2, \ldots\), and \(\gamma > 0\) is a smoothing parameter.

(b) Krylov methods are applied to solve problem (4), equivalently reformulated as

\[
\hat{x}_{k+1} = \arg \min_{\hat{x}} \|A S_{\gamma, k}^{-1} \hat{x} - b\|_2^2 + \frac{\lambda p}{2} \|\hat{x}\|_2^2,
\]

where the matrix \(S_{\gamma, k}^{-1}\) formally acts as a preconditioner for \(A\) and can be conveniently defined since \((W_{\gamma, k})_k\) is diagonal. In this setting, Krylov methods are much more efficient than the gradient descent methods currently used for NNR problems.

(c) The regularization parameter \(\lambda\) can be conveniently set during the inner iterations, using well-established parameter choice strategies that can be applied within Krylov solvers (see [2]). This is a clear upside with respect to both the available low-rank Krylov-like methods and NNR solvers, which require an appropriate value of the truncation rank \(\rho\) and the regularization parameter \(\lambda\), respectively, to be available in advance of the iterations (alternatively relying on the repeated application of such solvers for many values of \(\rho\) and \(\lambda\), respectively, to choose the most suitable ones).

(d) A solid theoretical analysis is provided for the new algorithms, which are regarded as particular instances of fixed-point iterations applied to an equivalent reformulation of the NNR problem. To the best of our knowledge, our new algorithms are the only principled strategies that employ Krylov methods for solving NNR problems, which may also be used in connection with well-posed problems.

Numerical experiments show that our new solvers are competitive with other state-of-the-art solvers for low-rank problems, and deliver reconstructions of increased quality with respect to classical Krylov methods. We refer to [1] for more details, including links to some MATLAB software.

References


Contour Integral Methods for Nonlinear Eigenvalue Problems: A Systems Theoretic Approach

Michael C. Brennan, Mark Embree, and Serkan Gugercin

Department of Mathematics, Virginia Tech, Blacksburg, VA, 24061, USA
https://intranet.math.vt.edu/people/gugercin/

Abstract

Let $T(z) : \mathbb{C} \to \mathbb{C}^{n \times n}$ denote an analytic matrix-valued function. The goal of the nonlinear eigenvalue problem (NLEVP) is to find $\lambda \in \mathbb{C}$ and a nonzero $v \in \mathbb{C}^n$ such that $T(\lambda)v = 0$. In this talk, we are not interested in finding all the eigenpairs of $T$; indeed there may be infinitely many. Instead, we are interested in solving the NLEVP in a compact region $\Omega \subset \mathbb{C}$. The starting point for the contour integral methods for solving this NLEVP is the following result from Keldysh [4].

**Theorem 1** Suppose $T(z) : \mathbb{C} \to \mathbb{C}^{n \times n}$ has $m$ eigenvalues $\lambda_1, \ldots, \lambda_m$ (counting multiplicity) in the region $\Omega \subset \mathbb{C}$, all semisimple. Then

$$T(z)^{-1} = V(zI - \Lambda)^{-1}W^* + N(z), \quad (1)$$

where $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_m)$, $V = [v_1 \cdots v_m] \in \mathbb{C}^{n \times m}$ and $W = [w_1 \cdots w_m] \in \mathbb{C}^{n \times m}$ contain the right and left eigenvectors $v_k$ and $w_k$ for the eigenvalue $\lambda_k$, normalized so $w_j^*T(\lambda_j)v_j = 1$; the matrix-valued function $N(z)$ is analytic in $\Omega$.

Based on Theorem 1, contour integration methods, as proposed in [2, 3], proceed by computing the quantities

$$A_k := \frac{1}{2\pi i} \int_{\partial \Omega} z^k L^*T(z)^{-1}R \, dz = \frac{1}{2\pi i} \int_{\partial \Omega} z^k L^*T(z)^{-1}R \, dz \in \mathbb{C}^{\ell \times p}, \quad (2)$$

where $L \in \mathbb{C}^{n \times \ell}$ and $R \in \mathbb{C}^{n \times p}$ are called the probing matrices. In an algorithmic setting, $A_k$ is computed via quadrature; we skip those details here. Once $A_k$ is computed for $k = 0, 1, 2, \ldots, K$, two Hankel matrices are constructed, from which the eigenvalues and eigenvectors of $T$ inside $\Omega$ are extracted.

In this talk, we will show that these contour integration techniques are closely related to data-driven system identification techniques from control theory, and illustrate how this systems theoretic perspective will expand the analysis and implementation of contour integration methods.

Our starting point is to realize $V(zI - \Lambda)^{-1}W^*$ in (1) as the transfer function of the multi-input/multi-output linear dynamical system

$$x'(t) = \Lambda x(t) + W^*u(t) \quad y(t) = Vx(t), \quad (3)$$

where $x(t) \in \mathbb{C}^m$, $u(t) \in \mathbb{C}^n$, and $y(t) \in \mathbb{C}^n$ are the states, inputs, and outputs of the dynamical system (3); see, e.g., [1, 6]. Let $U(z)$ and $Y(z)$ denote the Laplace transforms of $u(t)$ and $y(t)$. Then the transfer function $H(z) = V(zI - \Lambda)^{-1}W^*$ satisfies $Y(z) = H(z)U(z)$.

Expand $H(z)$ around $z = \infty$ to obtain

$$H(z) = V(zI - \Lambda)^{-1}W^* = \sum_{k=1}^{\infty} V \Lambda^{k-1} W^* z^{-k}.$$
The quantity $V \Lambda^{k-1} W^*$ is called the $k$th Markov parameter of the dynamical system (3). Therefore, in systems theoretic language, the quantity $A_k$ in (2) is the probed (tangentially observed) Markov parameter of $H(z)$. This observation reveals that contour integration methods for the NLEVP correspond to first sampling the Markov parameters of the transfer function $H(z) = V (zI - \Lambda)^{-1} W^*$ in (1) via contour integration of $T(z)^{-1}$, and then trying to recover (realize) $H(z)$ from these samples. The rational function $H(z)$ is the transfer function of the dynamical system corresponding to the underlying linear eigenvalue problem in the region of interest, $\Omega$.

This task is precisely what the Loewner framework [5] achieves in data-driven modeling of dynamical systems. In this talk, we will make these connections precise. Interpreting the contour integration methods as computing the Markov parameters of $H(z)$, i.e., the expansion coefficients around $z = \infty$, immediately suggests expanding $H(z)$ around finite points $\sigma \notin \Omega$ and using these expansion coefficients, called the moments of $H(z)$ around $z = \sigma$, in the setting of data-driven rational approximation to recover $H(z)$. Then, via a simple pole-residue expansion of $H(z)$, one can recover the desired eigenvalues and eigenvectors $T(z)$.

This new framework corresponds to replacing the polynomial term $z^k$ in (2) with a rational function $1/(z - \sigma)^k$. We will show that one does not need to sample at a single point $z = \sigma$ and can choose multiple interpolation points $\sigma_1, \sigma_2, \ldots, \sigma_r \notin \Omega$ to exactly recover the dynamical system $H(z)$, corresponding to the underlying linear eigenvalue problem in $\Omega$. We will also show that treating the problem as a system realization problem and employing the concept of tangential interpolation allows exact recovery of the eigenvectors and eigenvalues in $\Omega$, even in the case of two-sided probing. Various numerical examples will be used to illustrate the theoretical considerations.

References


Probabilistic Numerical Linear Solvers

Jon Cockayne, Ilse C.F. Ipsen, Chris J. Oates and Tim Reid

Department of Mathematics, North Carolina State University
https://ipsen.math.ncsu.edu

Abstract

We formulate iterative methods for the solution of nonsingular linear systems \( Ax = b \) as statistical inference processes by modeling the epistemic uncertainty in the iterates \( x_m \), that is, the limited knowledge about the solution \( x^* \) after \( m \) iterations. The goal is to obtain a well-calibrated uncertainty quantification (UQ) that is more insightful than traditional worst-case bounds, and to produce a probabilistic description of the error that can be propagated coherently through a computational pipeline.

The field of probabilistic numerics seeks a statistical treatment of the errors in deterministic numerical methods, such as numerical quadrature and the numerical solution of ordinary and partial differential equations. Its origins can be traced back to Poincaré [5], while a rigorous modern perspective is established in [3]. Applying the probabilistic numerical approach to the iterative solution of linear systems yields Bayesian linear solvers that posit a prior distribution for the quantity of interest, which can be the solution [2] or the matrix inverse [1, 4], and condition on the finite amount of information obtained during \( m \) iterations to produce a posterior distribution that reflects the reduced uncertainty after \( m \) iterations.

More specifically, the Bayesian Conjugate Gradient method (BayesCG) [2] for real symmetric positive-definite systems \( Ax = b \) models the uncertainty in \( x^* \) through a random variable \( X \sim N(x_0, \Sigma_0) \) drawn from a user-specified Gaussian prior with mean \( x_0 \) and covariance \( \Sigma_0 \). Given data in the form of search directions \( S_m \), there exists a unique Bayesian probabilistic numerical method [3] that conditions on \( y = S_m^T b \), and outputs a Gaussian posterior \( X | y \sim N(x_m, \Sigma_m) \) with

\[
\begin{align*}
x_m &= x_0 + \Sigma_0 A S_m \left( S_m^T \Sigma_0 A S_m \right)^{-1} S_m^T r_0 \\
\Sigma_m &= \Sigma_0 - \Sigma_0 A S_m \left( S_m^T \Sigma_0 A S_m \right)^{-1} S_m^T A \Sigma_0,
\end{align*}
\]

where \( r_0 = b - A x_0 \) is the initial residual. Here, the means \( x_m \) coincide with the approximations to the solution. Although one might object to Gaussians as a proper way of modeling uncertainty, they have the advantage of conjugacy, which assures a closed-form expression for the posterior, again as a Gaussian.

Requiring the search directions \( S_m \) to be \( A \Sigma_0 A \)-orthonormal obviates the necessity of inversion in (\(*\)). Then the posterior mean \( x_m \) lives in the Krylov space \( K_m^* \equiv x_0 + K_m(\Sigma_0 A^2, \Sigma_0 A r_0) \) and minimizes the error \( \| x - x_* \|_{\Sigma_0^{-1}} \) over \( K_m^* \). Expressing (\(*\)) as recursions,

\[
x_j = x_{j-1} - (s_j^T r_0) (\Sigma_{j-1} A s_j), \quad \Sigma_j = \Sigma_{j-1} - (\Sigma_{j-1} A s_j)(\Sigma_{j-1} A s_j)^T, \quad 1 \leq j \leq m,
\]

gives our iterative version of the BayesCG algorithm. BayesCG can be be viewed as preconditioning CG by \( \Sigma_0 A \). The expensive prior \( \Sigma_0 = A^{-1} \) recovers ordinary CG, while the even less affordable “natural covariance” \( \Sigma_0 = A^{-2} \) delivers instantaneous convergence.

For the posterior distribution to be useful, the UQ must be well-calibrated, which means the solution \( x_* \) should lie in a region where most of the posterior mass is located. If indeed the UQ...
were well-calibrated, then we could reasonably expect the solution to be drawn from the posterior: \( x_* \sim \mathcal{N}(x_m, \Sigma_m) \). This, in turn, would imply that a properly weighted error should be chi-squared distributed with \( d - m \) degrees of freedom, that is, \( Z \equiv \| (\Sigma_m^{1/2})^\dagger (x_* - x_m) \|^2 \sim \chi^2_{d-m} \). When used as a test statistic, and compared to the ideal distribution \( \chi^2_{d-m} \), the \( Z \)-statistic turns out to be overly conservative in numerical experiments, suggesting poorly calibrated UQ. The reason is the nonlinear dependence of CG on the solution \( x_* \) through the search directions \( S_m \) which depend on \( b = Ax_* \).

As time permits, we will address the following topics:

(i) Choice of prior: We want an informative prior that produces fast BayesCG convergence and is cheap to compute. The preconditioner prior \( P^{-2} \), where \( P = LL^T \) is an incomplete Cholesky IC(0) factorization of \( A \), approximates the natural covariance. Numerical experiments suggest much faster convergence than standard CG for the posterior means \( x_m \). However, the UQ is poorly calibrated: Although \( x_* \) is in the middle of the posterior, its spread is too large. As an alternative, we introduce the class of “Krylov priors” \( \Gamma_0 \) which promise fast convergence and well-calibrated UQ. The eigenvectors of a Krylov prior with \( \text{rank}(\Gamma_0) = k \) represent an \( A \)-orthonormal basis for \( \mathcal{K}_k(A, r_0) \). The ideal Krylov prior \( \Gamma_0 \), with maximal rank and properly chosen eigenvalues, produces posteriors \( \Gamma_m \) that have a clear relation to the true error,

\[
\| x_* - x_m \|^2_A = \text{trace}(A \Gamma_m).
\]

In practice, one can afford only priors \( \Gamma_0 \) of low rank, and such degenerate priors can pose statistical and numerical problems.

(ii) Error estimation: We introduce a statistic that appears to be less sensitive to the above mentioned nonlinear dependence on the solution: For independent random variables \( X, Y \sim \mathcal{N}(x_m, \Sigma_m) \) drawn from the posterior, the \( S \)-statistic \( S(X, Y) \equiv \| X - Y \|^2_A \) is a random variable with expectation

\[
\mathbb{E}[S(X, Y)] = 2 \text{trace}(A \Sigma_m).
\]

As with the \( Z \)-statistic we can then argue: If the UQ were well calibrated, then we could expect \( x_* \sim \mathcal{N}(x_m, \Sigma_m) \). This, in turn, implies that under a maximal-rank Krylov prior \( \Gamma_0 \), the \( S \)-statistic recovers the true error,

\[
\mathbb{E}[S(X, x_*)] = 2 \| x_m - x_* \|^2_A.
\]

(iii) Numerical stability: We are working on a suitable factored form for (***) that maintains semi-definiteness under down-dating. Orthogonality in finite precision depends on the conditioning of \( A \) and \( \Sigma_0 \). To keep the iteration count low, one can re-start BayesCG after \( k \) iterations with minimal disturbance because the most recent posterior \( \mathcal{N}(x_k, \Sigma_k) \) is the new prior.

References


Numerical Differential Geometry

Tingran Gao, Lek-Heng Lim, and Ke Ye

Department of Statistics, University of Chicago

www.stat.uchicago.edu/~lekheng/

Abstract

The subject that we call numerical differential geometry traces its roots to a classic paper of Edelman, Arias, and Smith [4]. One of their key insights is that certain Riemannian manifolds may be given extrinsic coordinates that allow points on the manifolds to be represented as matrices in a way that yields (a) closed-form analytic expressions for various differential geometric objects and operations — Riemannian metric, tangent space, geodesic, geodesic distance, parallel transport, Riemannian gradient and Hessian, etc. An immediate consequence is that (b) optimization algorithms on these “matrix manifolds” then require only standard numerical linear algebra, i.e., no need for numerical solutions of differential or nonlinear algebraic equations. This line of work has found numerous applications and attracted widespread interests; among other things, it has established manifold optimization [1] as a mainstay of optimization theory.

Properties (a) and (b) are not merely desirable; they are essential if one wants to realistically carry out computations on manifolds. Nevertheless, since the appearance of [4] twenty years ago, essentially only three matrix manifolds with the properties (a) and (b) have been discovered — the Stiefel manifold, the Grassmannian, and the positive definite cone, each equipped with its natural Riemannian metric. In our talk, we will show that this list may be vastly expanded:

1. **RIEMANNIAN:** We have recently extended the list of Riemannian manifolds with properties (a) and (b) to the Grassmannian of affine subspaces [7, 8] and flag manifold\(^\text{1}\) [10]. In an appropriate sense the latter includes Stiefel (flags of length zero) and Grassmann (flags of length one) manifolds as special cases and [10] may be viewed as a generalization of [4] to include flags of arbitrary lengths.

2. **semi-RIEMANNIAN:** We found certain restrictions, e.g., the manifold must be a so-called geodesic orbit space, that limits the class of Riemannian manifolds with properties (a) and (b), which explains why there are so few of them. In [5] we expanded our scope by including semi-Riemannian manifolds. In particular, we determined closed-form analytic expressions and developed optimization algorithms for pseudospheres, pseudo-hyperbolic spaces, de Sitter and anti de Sitter spaces, indefinite Stiefel and Grassmann manifolds, indefinite Lie groups, and other semi-Riemannian matrix manifolds.

3. **SYMPLECTIC:** This is work-in-progress intended to be an analogue of [4, 5] where a symplectic form (an antisymmetric 2-tensor) takes the place of the semi-Riemannian metric (a symmetric 2-tensor), a Hamiltonian flow takes the place of the gradient flow, and a Lie algebra of symplectic vector fields plays the role of the Lie algebra in [10]. Unlike the work in [2], we are not seeking to understand accelerated gradient methods under the light of symplectic geometry but creating algorithms for optimization on a symplectic manifold.

We expect these undertakings to be useful in applications the same way [4] has been useful. The main practical contribution of [4] is that it allows one to work with orthonormal bases and subspaces

\(^{1}\)A flag is a nested sequence of subspaces of increasing dimensions.
as one would work with points in $\mathbb{R}^n$; and in particular to optimize over all orthonormal bases or all subspaces. In the same spirit, our work in [7, 8, 10] allows one to work with affine subspaces and flags as if they are just points in $\mathbb{R}^n$. These are ubiquitous in applications: many statistical methods, e.g., linear regression, error-in-variables regression, principal components analysis, support vector machines, involve searching for an affine subspace that either best represents the data or best separates it into classes; likewise many numerical methods, e.g., finite-element, multigrid, spectral and pseudospectral, Krylov subspace, wavelet, involve seeking a nested sequence of subspaces, i.e., a flag, that provide increasingly better approximations to some true solution.

Another use of our work is in the design of optimization algorithms. As we noted in [5], optimizing a function $f : M \to \mathbb{R}$ on a smooth manifold $M$ is a problem that is independent of whatever additional structure on $M$, be that structure Riemannian, semi-Riemannian, symplectic, or something else — the simple reason being that the critical points of $f$ are independent of these structures. However imposing such structures gives us new algorithms for finding the critical points. Existing algorithms in optimization — steepest descent, conjugate gradient, accelerated gradient, Newton, quasi-Newton, etc — by-and-large rely only on a Riemannian structure and this sets an unnecessary boundary on developments of truly novel algorithms. For instance, one is hard-pressed to find an algorithm in continuous optimization that does not rely on the gradient of $f$ is some manner, although a notable exception is [3], which exploits the affine structure of a manifold instead of the Riemannian structure. One of our contributions is to show that semi-Riemannian (in [5]), symplectic (ongoing work), and other differential geometric structures on manifolds will similarly lead us to new optimization algorithms.

Aside from optimization, we will also discuss how considerations in numerical differential geometry give rise to new methods for defining and computing distances between subspaces of different dimensions [9] or covariance matrices of different dimensions [6].

Time permitting, we will discuss yet other differential geometric structures — affine, contact, Finsler — on matrix manifolds. Our long term goal is to develop a concrete version of differential geometry where every object can be represented as matrices and every operation can be computed with numerical linear algebra — it is what we envision to be the core of numerical differential geometry, on which more sophisticated algorithms and methods can be built.

References


Block Krylov Subspace Methods for Functions of Matrices II: Modified block FOM

Andreas Frommer, Kathryn Lund, and Daniel B. Szyld

Department of Numerical Mathematics, Charles University
https://www.karlin.mff.cuni.cz/~lundka

Abstract

What began as a topic proposal for a doctoral thesis—“what if we use block Krylov subspace methods (KSMs) to compute matrix functions?”—turned into a journey through many strata of numerical linear algebra.

Using KSMs to compute a matrix function acting on a vector \( f(A)b \) is not a novel idea; see, e.g., [2, 10, 17, 20]. Block KSMs are even older, and have been used since the 1980s for solving linear systems with multiple right-hand sides [14, 21]. At first glance, the answer to our question seems like a simple substitution of block KSMs into the matrix function literature, especially if we just want to compute \( f(A)B \), where \( B = [b_1 \cdots b_s] \) is a block vector. Although this is true in some respects, the analysis is nontrivial, as many properties taken for granted with column KSMs do not transfer to block KSMs.

Our journey started with block Full Orthogonalization Methods (FOM) for \( f(A)B \) [6] and two main discoveries: 1) many approaches are actually block methods that are not often regarded as such; and 2) restarting block KSMs is not only necessary, because we must store the full basis for the matrix function computation, but also more difficult, because what were before colinear (i.e., scalar) relationships between column vectors now become cospatial (i.e., matrix) relationships between the subspaces spanned by the columns of block vectors.

In response to our first discovery, we developed a framework for simultaneously treating a variety of block KSMs, differentiating mainly by how the scalar inner product is generalized to block vectors. The essential, “classical” inner product is given by \( X^*Y \), and other paradigms are variations on this theme: “global” methods average the diagonal entries of \( X^*Y \) [3, 11]; “loop-interchange” methods extract only the diagonal of \( X^*Y \) [16]; and “hybrid” methods extract block diagonal entries of \( X^*Y \) for a specified block size. Within such a framework, we can talk also connect computational costs for different block inner products on the implementation level with convergence behavior for the associated block Krylov subspace approximations on the theoretical level. In our present work [7], we prove precise comparison statements on this front for not just FOM-type methods, but also for generalized minimal residual (GMRES) methods and Radau-Arnoldi (RA) methods, which allow for prescribing Ritz values in the spirit of Gauß-Radau quadrature rules [5, 8].

With regards to restarting, we focus on functions with Riemann-Stieltjes representations,

\[
    f : \mathbb{C} \setminus (-\infty, 0] \to \mathbb{C}, \quad f(z) = \int_{0}^{\infty} \frac{1}{z + t} \, d\mu(t),
\]

allowing us to reduce the analysis of block KSMs for matrix functions to integrals over families of shifted systems:

\[
    f(A)B = \int_{0}^{\infty} (A + tI)^{-1}B \, d\mu(t) \approx \int_{0}^{\infty} X_m(t) \, d\mu(t), \quad (1)
\]

where \( X_m(t) \) approximates the solution to

\[
    (A + tI)X(t) = B. \quad (2)
\]
Since the Arnoldi relation is shift-invariant, only one Krylov subspace is needed to build approximations for (2). Furthermore, when \( \mathbf{X}_m(t) \) is a family of block FOM approximations, the intuition from column FOM [18] transfers quite nicely: residuals from shifted linear systems turn out to be “naturally” cospatial:

\[
\mathbf{R}_m(t) = \mathbf{B} - (A + tI)\mathbf{X}_m(t) = \mathbf{R}_m(0)C_m(t), \text{ where } C_m : [0, \infty) \to \mathbb{C}^{n \times s}.
\]  

(3)

Formulating a restarting procedure for approximations to (1) therefore hinges on tracking formulas for the cospatial factors \( C_m(t) \) from previous restart cycles.

When \( \mathbf{X}_m(t) \) are instead GMRES-type approximations, we lose automatic cospatiality. By demanding that all initial approximations \( \mathbf{X}_0(t) \) be identical, we can recover cospatial statements of the form (3), but the approximations \( \mathbf{X}_m(t) \) are then no longer necessarily residual-minimizing. The properties of \( C_m(t) \) also become more difficult to understand. In the column case [4], the colinear factor can be written in terms of a scalar polynomial whose roots are harmonic Ritz values;\(^1\) this fact allows one to bound the shifted residuals by that of the seed system, whenever \( A \) is positive real.\(^2\) In the block case, although we do have that \( C_m(t) = P(-t)^{-1} \), where \( P \) is a matrix polynomial\(^3\) whose latent roots\(^4\) are harmonic Ritz values, we can only guarantee that \( \det(C_m(t)) \leq 1 \). Indeed, determining the conditions under which \( \mathbf{R}_m(t) \) is bounded for block GMRES and positive real \( A \) remains an open and challenging question.

An underlying theme of our work stems from a generalization of the polynomial exactness result for KSMs [5, Lemmas 1.3 & 1.4], along with an alternative characterization of block GMRES [19, Theorem 3.3]. In simplest terms, if we have a block KSM with the Arnoldi relation

\[
AV_m = \mathbf{V}_m\mathbf{H}_m + \mathbf{V}_{m+1}\mathbf{H}_{m+1,\mathbf{E}_m^*},
\]  

(4)

then for any matrix polynomial \( Q \) of degree at most \( m - 1 \) and any block vector \( \mathbf{M} \in \mathbb{C}^{n \times s} \), it holds that \( \mathbf{V}_mQ(\mathbf{H}_m + \mathbf{M}\mathbf{E}_m^*) \circ (\mathbf{V}_m\mathbf{B}) \) is an element of the block Krylov subspace and therefore a candidate approximation to \( A\mathbf{X} = \mathbf{B} \). Both block GMRES and block RA approximations can be expressed as specific low-rank alterations \( \mathbf{M}\mathbf{E}_m^* \) to \( \mathbf{H}_m \), thus rendering them types of “modified block FOM.” This result, paired with matrix polynomial theory, leads to a richer understanding of Ritz values and harmonic Ritz values for block methods. Thanks to RA methods specifically, we are even able to prescribe some Ritz values, and numerical results indicate there is potential to reduce iteration counts compared to FOM-type and GMRES-type methods for matrix functions when the extremes of the spectrum of \( A \) are accessible.

What began as a one-sentence curiosity transformed into a generalized framework for block KSMs, particularly modified block FOM, as well as a richer understanding of their applications to linear systems, families of shifted systems, and matrix functions.\(^5\) Our work also motivates new research directions. Numerical experiments in [7] raise questions about how to select a block inner product, particularly in scenarios where global methods take as many iterations as classical ones to converge to the same tolerance. With matrix function computations generally being more costly than those of linear systems, this choice begs careful consideration. Preliminary work further shows the potential of our framework in understanding and constructing communication-reducing methods in high-performance computing (such as, e.g., [1, 9]), and a Löwner-ordering approach to our framework [12] may prove key for resolving the open question of bounding shifted GMRES-type residuals.

---

\(^1\)Harmonic Ritz values are the inverses of the ordinary Ritz values of \( A^{-1} \) in \( \mathbb{K}_m \); see [15].

\(^2\)\( A \) is positive real if its field of values belongs to the right half of the complex plan.

\(^3\)Here we mean polynomials whose coefficients are \( s \times s \) matrices.

\(^4\)The latent roots of a matrix polynomial \( P \) are the roots of the polynomial \( \det(P) \).

\(^5\)It also achieved its original goal of transforming into a thesis [13].
References


Abstract

James Wilkinson developed a systematic way of analyzing rounding errors in numerical algorithms in the 1960s—a time when computers typically used double precision arithmetic and a matrix of dimension $n = 100$ was considered large. As a result, an error bound with a dominant term $p(n)u$, for $p$ a low degree polynomial and $u$ the machine precision, was acceptably small.

Today, the supercomputers heading the TOP500 list solve linear systems of equations of dimension $10^8$ and half precision arithmetic (accurate to about 4 significant decimal digits) is increasingly available in hardware, notably on graphical processing units (GPUs) from AMD and NVIDIA. Traditional rounding error bounds cannot guarantee accurate results when the dimension is very large or the precision is low.

The current absence of stability guarantees for large scale, low precision computations has therefore created a need for new error bounds that are sharper on average. Indeed, traditional bounds tend to be pessimistic: while the composition of $n$ operations may lead to an error bound proportional to $nu$, this worst case is attained only when each rounding error is of maximal magnitude and identical sign, which is very unlikely.

Since the beginning of the digital computer era many researchers have modelled rounding errors as random variables in an attempt to account for these statistical effects and obtain better estimates of how the error behaves on average. This line of thinking has led to the well-known rule of thumb that constants in rounding error bounds can be replaced by their square roots. However, this rule of thumb has never been rigorously proven for a wide class of numerical algorithms.

In this talk, we will present a new approach to probabilistic rounding error analysis that provides the first rigorous justification of the rule of thumb [1]. In the same vein as Wilkinson’s backward error analysis, our analysis aims at being systematic by analyzing a general term of the form $\theta_n = 1 - \prod_{i=1}^n (1 + \delta_i)$, where $|\delta_i| \leq u$. Traditionally, $|\theta_n|$ is bounded by $\gamma_n = nu/(1 - nu)$, an ubiquitous constant in error analysis. Based on a probabilistic model of the rounding errors that assumes the $\delta_i$ to be independent random variables of mean zero, we prove that $|\theta_n|$ can be bounded by a relaxed constant $\tilde{\gamma}_n$ which is a small multiple of $\sqrt{nu}$ to first order. Our analysis can therefore be systematically applied to any numerical algorithm, including the solution of linear systems, by simply replacing $\gamma_n$ by $\tilde{\gamma}_n$ in the error bound.

More precisely, the bound $|\theta_n| \leq \lambda \sqrt{nu}$ holds with a given probability $P(\lambda)$ which grows exponentially close to 1 as $\lambda$ increases. Thus, small values of $\lambda$ (say, less than 10) suffice to make the bounds hold with probability close to 1. Our analysis is therefore able to maintain stability guarantees, in a probabilistic sense, for large scale and low precision computations for which the traditional analysis cannot guarantee even a single correct digit.

We support our analysis with some numerical experiments on important numerical linear algebra kernels such as matrix–vector multiplication and the solution of linear systems by LU factorization. The results show that our probabilistic bounds are often in excellent agreement with the
actual backward error for a wide range of both random and real-life matrices coming from various applications.

Our experiments also reveal a strong dependence of the backward error on the data distribution. In particular, when using random matrices with uniform entries sampled in $[-1, 1]$, the backward error remains of order $u$ even for large $n$. In this case, even the probabilistic bound proportional to $\sqrt{n}u$ is therefore pessimistic.

In order to explain this phenomenon, we perform a new analysis [2]. We combine the probabilistic model of rounding errors with a second probabilistic model of the data, and prove the observed behavior of the backward error is related to the mean of the data. For data with zero or very small mean, we obtain sharper backward error bounds of the form $cu$, where $c$ is a modest constant independent of $n$.

References


Spurious Eigenvalues of the Linearization of Nonlinear Eigenvalue Problems

Karl Meerbergen

Department of Computer Science, KU Leuven
people.cs.kuleuven.be/karl.meerbergen

Abstract

This work is related to the linearization of nonlinear matrix valued functions

\[ A : \mathbb{C} \rightarrow \mathbb{C}^{n \times n}. \]

The nonlinear eigenvalue problem is an important application, i.e., finding the roots \( \lambda \) of

\[ \det A(\lambda) = 0 \]

in a region \( \Sigma \subset \mathbb{C} \). An example is the finite element model of a beam of insulated material sandwiched between two steel plates:

\[ A(s) = K + s^2 M + \frac{G_0 + G_\infty (s\tau)^\alpha}{1 + (s\tau)^\alpha} C, \]

where \( s = \omega \) with \( \omega \) the frequency.

When \( A \) is a matrix polynomial (or a rational matrix valued function), there are linear pencils \( A - sB \in \mathbb{C}^{n\times n \times n} \) and unimodular (i.e. with constant determinant) \( E,F : \mathbb{C} \rightarrow \mathbb{C}^{n \times n} \) such that

\[
\begin{bmatrix}
A(s) & 0 \\
0 & I
\end{bmatrix} = E(s)(A - sB)F(s) \text{ for } s \in \mathbb{C}.
\]

Matrix \( A - sB \) is called a linearization of \( A(s) \). This relation is very useful for computing eigenvalues of \( A(s) \) using numerical methods for linear eigenvalue problems on \( A - sB \). It also has applications in model order reduction, e.g., of second order systems.

For nonlinear, nonpolynomial, \( A \), linearizations exist that provide an approximation for \( s \) in a region \( \Sigma \):

\[
\begin{bmatrix}
A(s) & 0 \\
0 & I
\end{bmatrix} \approx E(s)(A - sB)F(s) \text{ for } s \in \Sigma.
\]

When the approximation is accurate, the linearization can be used for computing eigenvalues. The last decade several approximation techniques were proposed: a polynomial approximation using Chebyshev polynomials [1], a rational approximation using a spectral discretization (infinite Arnoldi) [3], a Padé approximation [6], a rational approximation from potential theory (NLEIGS) [2], and a rational approximation using AAA [5].

Potential theory is used in NLEIGS for determining a rational approximation of \( A \). The rational approximation minimizes the error of \( A(s) \) on \( \Sigma \) and choses poles on a user defined singularity set \( \Xi \), which is usually chosen as the branch cuts of the coefficients of \( A(s) \). AAA determines the rational approximation by minimizing the approximation error of \( A(s) \) on a discretization of \( \Sigma \). No singularity set should be provided. For all our examples, AAA produces a lower degree approximation than potential theory.

Next to eigenvalue computations, linearizations are also used to simulate mechanical components described by nonlinear matrix valued functions in the time domain. Nonlinear eigenvalue problems
also arise in the stability analysis of various kinds of applications. In both cases, it is important that there are no spurious eigenvalues on the right of the spectrum, since this may lead to a wrong determination of stability or instabilities during time integration. Preservation of the property of stability is therefore an important requirement of a linearization. Unfortunately, the rational approximations often do not preserve stability of the original system.

We will present several approaches that reduce or eliminate the chance of instable eigenvalues. Firstly, we define appropriate choices of $\Sigma$ and $\Xi$ for AAA and potential theory. The set $\Sigma$ corresponds to the region of interest where $A(s)$ is approximated. For dynamical systems, this is often a region along an interval on the frequency axis, i.e., the imaginary axis for $s$. The model is only valid for a limited frequency range. For reasons of efficiency, there is no need to approximate $A(s)$ well for $s$ outside this interval. It appears that, when AAA is used as approximation method, enlarging $\Sigma$ significantly reduces the spurious eigenvalues in the right half plane, or pushes them away towards eigenvalues with large modulus. Achieving this seems to be more difficult with potential theory. We noticed that one advantage of potential theory is the freedom in choosing the singularity set. But, unfortunately, none of the linearizations based on potential theory or AAA are guaranteed to eliminate spurious eigenvalues.

Secondly, we will show results for other linearizations. These include approximations obtained using methods from model order reduction. We also exploit the fact that the eigenvalues of

$$A(s) = K + s^2 M + \sum_{j=1}^{d} C_j \frac{s\tau_j}{1 + s\tau_j}$$

are stable with $K$, $M$ and $C_1, \ldots, C_d$ symmetric positive definite, if $\tau_1, \ldots, \tau_d$ are real and positive. We will explain how the coefficients can be matched to obtained a good approximation.

The conclusion of the story is that many questions remain about reliable approximations of $A(s)$ arising from dynamical systems.

References


An Interpolatory Subspace Framework for Nonlinear Eigenvalue Problems

Emre Mengi

Department of Mathematics, Koç University, Istanbul, Turkey
http://home.ku.edu.tr/~emengi

Abstract

The rational eigenvalue problems of the form

\[ R(\lambda)v = 0 \quad \exists v \in \mathbb{C}^n \setminus \{0\}, \quad \text{where} \quad R(s) := P(s) + \sum_{j=1}^{\kappa} \frac{p_j(s)}{d_j(s)} E_j, \quad (1) \]

arise from various applications such as the modeling of vibrations of fluid-solid structure, as well as vibrating mechanical structures [1]. Here, \( E_0, \ldots, E_\kappa \in \mathbb{C}^{n \times n} \) are given matrices, whereas \( P(s) := s^d P_d + \cdots + s P_1 + P_0 \) with \( P_0, \ldots, P_d \in \mathbb{C}^{n \times n} \) is a given matrix polynomial, and \( p_j, d_j : \mathbb{C} \to \mathbb{C} \) are scalar polynomials such that the degree of \( d_j \) is greater than that of \( p_j \).

Based on the observation that the proper rational part of \( R(s) \) can be expressed as

\[ R_p(s) := \sum_{j=1}^{\kappa} \frac{p_j(s)}{d_j(s)} E_j = C(A - sI)^{-1}B \quad (2) \]

for some \( C \in \mathbb{C}^{n \times N}, A \in \mathbb{C}^{N \times N}, B \in \mathbb{C}^{N \times n} \) and \( N := \sum_{j=1}^{\kappa} \text{Rank}(E_j) \cdot \text{degree}(d_j) \), we first propose a subspace framework to compute the eigenvalues of \( R_p(s) \) closest to a prescribed target \( \tau \in \mathbb{C} \). Then we extend the proposed subspace framework to an arbitrary nonlinear eigenvalue problem, including the rational eigenvalue problem in (1).

Proper Rational Eigenvalue Problems. In the setting of (2), we assume that the middle factor \( A \) is large, more specifically we assume \( N \gg n \). This is in particular the case when the summation in (2) involves many terms. As a remedy, in our subspace framework, \( R_p(s) \) is replaced by

\[ R_p^{W,V}(s) := CV(sW^*V - W^*AV)^{-1}W^*B \]

for two subspaces \( W, V \) of equal dimension, say \( r \) such that \( r \ll N \), where \( W, V \) denote \( N \times r \) matrices whose columns form orthonormal bases for the subspaces \( W, V \). Note that \( r \times r \) middle factor of the reduced proper rational function \( R_p^{W,V}(s) \) is much smaller than that of the full problem.

We compute the eigenvalue \( \tilde{\lambda} \) of \( R_p^{W,V}(s) \) closest to the target point \( \tau \). Then we expand the subspaces \( W, V \) to \( \tilde{W}, \tilde{V} \) so that

\[ R_p(\tilde{\lambda}) = R_p^{\tilde{W},\tilde{V}}(\tilde{\lambda}) \quad \text{and} \quad R_p^{(j)}(\tilde{\lambda}) = \left[ R_p^{\tilde{W},\tilde{V}} \right]^{(j)}(\tilde{\lambda}), \quad j = 1, \ldots, M \quad (3) \]

for a prescribed positive integer \( M \), where \( R_p^{(j)}(\tilde{\lambda}) \) and \( \left[ R_p^{\tilde{W},\tilde{V}} \right]^{(j)}(\tilde{\lambda}) \) denote the \( j \)-th derivatives of \( R_p(s) \) and \( R_p^{\tilde{W},\tilde{V}}(s) \) at \( \tilde{\lambda} \). The procedure is repeated after replacing \( W, V \) with \( \tilde{W}, \tilde{V} \).

Two important details are in order. First, the finite eigenvalues of \( R_p^{W,V}(s) \) are the same as those of the pencil

\[ \mathcal{L}^{W,V}(s) := A^{W,V} - sB^{W,V}, \quad \text{where} \quad A^{W,V} := \begin{bmatrix} W^*AV & W^*B \\ CV & 0 \end{bmatrix}, \quad B^{W,V} := \begin{bmatrix} W^*V & 0 \\ 0 & 0 \end{bmatrix}, \]

\[ \mathcal{L}^{W,V}(s) := A^{W,V} - sB^{W,V}, \quad \text{where} \quad A^{W,V} := \begin{bmatrix} W^*AV & W^*B \\ CV & 0 \end{bmatrix}, \quad B^{W,V} := \begin{bmatrix} W^*V & 0 \\ 0 & 0 \end{bmatrix}, \]

\[ \mathcal{L}^{W,V}(s) := A^{W,V} - sB^{W,V}, \quad \text{where} \quad A^{W,V} := \begin{bmatrix} W^*AV & W^*B \\ CV & 0 \end{bmatrix}, \quad B^{W,V} := \begin{bmatrix} W^*V & 0 \\ 0 & 0 \end{bmatrix}, \]
excluding the ones that are also the eigenvalues of $W^*AV$. Hence, we retrieve $\tilde{\lambda}$ by computing all of the eigenvalues of $L^{W,V}(s)$ and choosing the one closest to the target point. Secondly, it is an immediate corollary of [2, Theorem 1] that the Hermite interpolation properties in (3) can be achieved by means of the inclusions $\tilde{\mathcal{V}} \supseteq \bigoplus_{j=1}^M \text{Ran } [(A - \tilde{\lambda}I)^{-j}B]$ and $\tilde{\mathcal{W}} \supseteq \bigoplus_{j=1}^M \text{Ran } [(C(A - \tilde{\lambda}I)^{-j})^*]$. 

**General Nonlinear Eigenvalue Problem Setting.** Consider the nonlinear eigenvalue problem

$$T(\lambda)v = 0 \quad \forall v \in \mathbb{C}^K \setminus \{0\}, \quad \text{where } T(s) := f_1(s)T_1 + \cdots + f_\kappa(s)T_\kappa,$$

where $f_1, \ldots, f_\kappa \in \mathbb{C} \to \mathbb{C}$ are meromorphic functions, and $T_1, \ldots, T_\kappa \in \mathbb{C}^{K \times K}$. This setting encompasses polynomial and delay eigenvalue problems, and many others. Inspired by the ideas for the proper rational eigenvalue problems, let us partition $T(s)$ and $T_j$ for $j = 1, \ldots, \kappa$ as

$$T(s) = \begin{bmatrix} A(s) & B(s) \\ C(s) & D(s) \end{bmatrix}, \quad T_j = \begin{bmatrix} A_j & B_j \\ C_j & D_j \end{bmatrix},$$

where $A(s), A_j \in \mathbb{C}^{N \times N}, B(s), B_j \in \mathbb{C}^{N \times n}, C(s), C_j \in \mathbb{C}^{n \times N}, D(s), D_j \in \mathbb{C}^{n \times n}$ such that $N \gg n$. Then, every finite eigenvalue $\lambda$ of $T(s)$, that is not an eigenvalue of $A(s)$, is also an eigenvalue of

$$\mathcal{R}(s) := C(s)A(s)^{-1}B(s) - D(s).$$

Corresponding to $\mathcal{R}(s)$, we define the reduced matrix-valued function

$$\mathcal{R}^{W,V}(s) := C^V(s)A^{W,V}(s)^{-1}B^W(s) - D^W(s),$$

where $A^{W,V}(s) := W*AV = \sum_{j=1}^\kappa f_j(s)(W^*A_jV)$, $B^W(s) := W*B(s) = \sum_{j=1}^\kappa f_j(s)(W^*B_j)$, and $C^V(s) := C(s)V = \sum_{j=1}^\kappa f_j(s)(C_jV)$. The eigenvalues of $\mathcal{R}^{W,V}(s)$ are the same as those of

$$T^{W,V}(s) = \begin{bmatrix} A^{W,V}(s) & B^W(s) \\ C^V(s) & D^W(s) \end{bmatrix},$$

except possibly those that are the eigenvalues $A^{W,V}(s)$.

Our subspace framework for given subspaces $\mathcal{V}, \mathcal{W}$ locates the eigenvalue $\tilde{\lambda}$ of $\mathcal{R}^{W,V}(s)$ closest to the target. This is again achieved by computing all of the eigenvalues of $T^{W,V}(s)$ and choosing the one closest to the target. Then the subspaces are expanded into $\tilde{\mathcal{W}}, \tilde{\mathcal{V}}$ so as to ensure $\mathcal{R}(\tilde{\lambda}) = \mathcal{R}^{\tilde{\mathcal{W}},\tilde{\mathcal{V}}}(\tilde{\lambda})$ and $\mathcal{R}^{(j)}(\tilde{\lambda}) = \left[\mathcal{R}^{\tilde{\mathcal{W}},\tilde{\mathcal{V}}}\right]^{(j)}(\tilde{\lambda})$ for $j = 1, \ldots, M$. The inclusions $\tilde{\mathcal{V}} \supseteq \bigoplus_{j=1}^M \text{Ran } [(\tilde{\lambda})^{-j}B(\tilde{\lambda})]$ and $\tilde{\mathcal{W}} \supseteq \bigoplus_{j=1}^M \text{Ran } [(C(\tilde{\lambda})\tilde{\lambda}^{-j})^*]$ guarantee the satisfaction of these interpolation properties.

**Convergence Properties.** In practice, we observe that the subspace frameworks above converge nearly always to the eigenvalue closest to the target point. The subspace frameworks can be generalized to locate $k$ nearest eigenvalues to the target by interpolating the original matrix-valued function at $k$ eigenvalues of the reduced problem closest to the specified target. In various cases, such as the proper rational functions when $A$ has a banded structure and polynomial eigenvalue problems, the proposed subspace frameworks run several times faster than `eigs` in Matlab.

In theory, denoting with $\lambda$ and $\tilde{\lambda}$ the current and the next eigenvalue estimates by the subspace frameworks, we show that $|\tilde{\lambda} - \lambda| \leq C|\lambda - \lambda_*|^2$ for some constant $C$, provided that $\lambda, \tilde{\lambda}$ are sufficiently close to an eigenvalue $\lambda_*$ and under some non-defectiveness assumptions on $\lambda_*$. 

67
References


Towards Randomized Eigensolvers for Exascale Computing

Agnieszka Miedlar

Department of Mathematics, University of Kansas
http://miedlar.faculty.ku.edu/

Abstract

With the arrival of peta- and exascale supercomputers, and modern scientific applications becoming more complex and data intensive ("data greedy"), with the cost of moving a single word already exceeding the cost of single arithmetic operation [1], current iterative solvers are not able to take full advantage of new architectures, and meet the expected efficiency and storage requirements. Therefore, fundamental research in numerical linear algebra is essential for the future of exascale computing.

Next generation algorithms for solving fundamental problems of linear algebra, i.e., linear systems, least-squares or eigenvalue problems, will need to significantly reduce the amount of data pulled into the memory, e.g. by examining only a subset of the input matrix (randomized algorithms [2]), minimize communication (communication-avoiding algorithms [3]), and allow decentralized processing and communication algorithms (asynchronous algorithms [4]). Of particular interest to us are randomized and communication-avoiding algorithms for eigenvalue computations. Single-pass randomized algorithms as well as communication-avoiding numerical methods for computing standard factorizations, e.g., SVD, QR or the eigenvalue decomposition etc., have been extensively studied in [2, 5, 6]. [7] presented parallel and sequential eigenvalue algorithms for dense symmetric/nonsymmetric matrices and matrix pencils that perform $O(n^3)$ arithmetic operations, are numerically stable, and attain the known communication lower bounds for $O(n^3)$ dense linear algebra algorithms [3]. Moreover, recent results on randomized linear iterative methods for solving linear systems [8] and eigenvalue problems [9], lay the foundation for new developments in randomized and communication-avoiding algorithms for efficiently solving challenging large-scale eigenvalue problems.

In this talk, we will first present the state-of-the-art results for solving dense eigenvalue problems. We will study the already existing algorithms which attain the theoretical communication lower bounds, e.g., randomized spectral divide-and-conquer algorithm for computing spectrum of a general large matrix (or matrix pencil) [10, 7] using (generalized) Schur decomposition [7] form and successive band reduction [7, 11]. We will discuss extensions of the existing theoretical and algorithmic work on communication-avoiding and randomized algorithms for the reduction to tridiagonal form to two-sided factorization which for a given matrix will enable its (generalized) upper Hessenberg form. Here, we will not only consider minimizing communication, but also optimizing the overall number of floating-point operations. Even though fast algorithms for determining the eigenvalues of the tridiagonal or banded matrices do exist [12, 11], the problem of finding the eigenvalues of a matrix in an upper Hessenberg form needs to be addressed. Based on the recent results establishing the relationship between randomized and deterministic rank-revealing QR factorization [13, 14], we will revisit the classical QR algorithm and the Hessenberg QR iteration for general matrices, establishing the communication lower bounds, developing the practical algorithms and determining their actual communication costs. All newly proposed algorithms will be illustrated with numerical experiments to verify their accuracy, stability and overall performance with respect to the conventional algorithms. At this point, it is important to remember, that the poor performance of existing algorithms on parallel computers and modern cache-based processors is not exclusive to direct methods. Due to sparse matrix-vector multiplications and frequent synchronization, even iterative methods are communication intensive and therefore, in their
present form, not able to efficiently exploit the novel architectures. Although research has been conducted in the area of communication-avoiding Krylov subspace methods [15, 16], most studies have focused on solving linear systems leaving many challenges in solving eigenvalue problems to be faced. Communication-avoiding Lanczos algorithms have been developed in [17, 18] for both symmetric and nonsymmetric eigenvalue problems. Minimizing communication is achieved by exploiting the BLAS 3 operations (e.g. block Gram-Schmidt and skinny QR) when performing sparse matrix-vector multiplications, and different reorthogonalization schemes (e.g. explicitly restarted Lanczos). On the other end, randomized algorithms for computing the dominant eigenmodes of the generalized Hermitian eigenvalue problems $Ax = \lambda Bx$, with matrices $A, B$ Hermitian and Hermitian positive definite, respectively, were proposed in [9]. Build upon this results and our prior work on contour integral eigenvalue solvers, we will present a randomized eigenvalue solver based on contour integration, i.e., the randomized FEAST algorithm.

References


Partial Smoothness of the Numerical Radius

Adrian S. Lewis and Michael L. Overton
Courant Institute of Mathematical Sciences, New York University
www.cs.nyu.edu/overton

Abstract

We start with a short summary. The numerical radius of a matrix is the maximum of the modulus of the elements in its field of values. We have observed that minimizing the numerical radius of a parametrized matrix often produces a solution in the class of “disk matrices”: those whose field of values is a disk in the complex plane centered at the origin. We explain this phenomenon by studying the “partial smoothness” of the numerical radius. Let \( r \) denote the numerical radius and let \( X \) be a fixed disk matrix. Using tools from convex analysis, we give conditions under which the set of all disk matrices near \( X \) form a manifold \( M \) with respect to which \( r \) is “partly smooth” at \( X \). Roughly speaking, this means that, when restricted to \( M \) near \( X \), \( r \) is continuously differentiable, but along directions transversal to \( M \), \( r \) is nonsmooth with a “V-shaped” graph. Thus, minimizing the numerical radius of a parametrized matrix often results in solutions lying on \( M \). In particular, we show that \( r \) is partly smooth at “superdiagonal matrices”: \( n \times n \) disk matrices with exactly \( n-1 \) nonzeros, all on the superdiagonal. Finally, we consider the general \( 3 \times 3 \) case; even then, the details are surprisingly intricate, with \( r \) partly smooth at most, but not all, \( 3 \times 3 \) disk matrices.

In what follows we begin by motivating our interest in the numerical radius, and then we give an extended summary of the proposed presentation. For algorithms to compute the numerical radius, see [13, 14]; an alternative approach using convex optimization (semidefinite programming) is based on a formulation due to Mathias [12].

Given an \( n \times n \) complex matrix \( A \), the asymptotic convergence of the discrete-time dynamical system \( x_{k+1} = Ax_k, k = 0, 1, 2, \ldots \), is governed by the spectrum of \( A \). Specifically, the iterates \( x_k \in \mathbb{C}^n \) converge to zero for all initial points \( x_0 \) if and only if \( A \) is stable, meaning its spectrum is contained in the open unit disk. However, to understand the transient behavior of the iterates — how \( \|x_k\| \) depends on \( k \) — the spectrum is inadequate. Instead we can exploit the following well-known inequality:

\[
\|A^k\|_2 \leq 2(r(A))^k, \quad k = 0, 1, 2, \ldots,
\]

where \( \| \cdot \|_2 \) denotes the spectral norm and \( r(A) \) denotes the numerical radius of \( A \),

\[
r(A) = \max \{ |u^*Au| : u \in \mathbb{C}^n, \|u\|_2 = 1 \}.
\]

The numerical radius is a vector space norm, but it does not satisfy the submultiplicative property required in most definitions of a matrix norm. However, Lax and Wendroff, in their famous 1964 paper on difference equations for hyperbolic PDEs [10], showed that if \( r(A) \leq 1 \), then the sequence \( \|A^k\| \) is bounded. Halmos then conjectured the “power inequality”

\[
r(A^k) \leq (r(A))^k, \quad k = 0, 1, 2, \ldots,
\]

which was subsequently proved by Berger [3]. The inequality (1) is an elementary consequence.

The numerical radius appears naturally in matrix optimization and best approximation problems: one survey, from an operator-theoretic perspective, appears in [1]. Our work is motivated by the observation that optimization problems involving the numerical radius often result in disk
matrices. Surprisingly, an explicit parametrization of disk matrices was not known until recent work of Crouzeix [6], which is based on much older work of Ando [2].

Disk matrices are relatively rare: we focus here primarily on $n \times n$ matrices with exactly $n - 1$ nonzeros, all on the superdiagonal, such as nilpotent Jordan blocks and the Crabb matrix [4] that arises in Crouzeix’s conjecture [5, 7, 9]. We show that, near these matrices, the set of disk matrices has the structure of a manifold $\mathcal{M}$ of codimension $2n$ in the $2n^2$-dimensional real vector space $\mathbb{C}^{n \times n}$. Our aim here is to illuminate variational properties of the numerical radius $r$, in particular the property of partial smoothness [11] that, notwithstanding their rarity, produces disk matrices as optimizers of the numerical radius $r$ over parametrized matrices. Specifically, we show that at superdiagonal matrices, the numerical radius is partly smooth with respect to $\mathcal{M}$. There is no room to define partial smoothness here, but we motivate it by a simple example: the function $(x, y) \mapsto (x - 1)^2 + |y - x^2|$ is partly smooth with respect to the manifold, in this case a parabola, $\{(x, y) : y = x^2\}$. The minimizer of this function, $(1, 1)$, lies on this manifold.

Our presentation will be organized as follows. We first discuss two interesting examples where disk matrices appear as solutions to optimization problems involving the numerical radius. The first example involves feedback control, while the second arises in Crouzeix’s conjecture. Then we introduce the ideas we use from convex analysis, including the notions of subgradients and partial smoothness. We illustrate these using the Crabb matrix, building on our earlier work with Greenbaum in [8]. We then give a general characterization of subgradients and partial smoothness of the numerical radius at disk matrices. Finally, we discuss our results in the framework of the Crouzeix parametrization of disk matrices, apply them to superdiagonal matrices, and conclude with a complete analysis of the general $3 \times 3$ case. In particular, we show that in $\mathbb{C}^{3 \times 3}$, a real vector space with dimension 18, the set of disk matrices is the closure of a semi-algebraic manifold with dimension 12.

References


Approximation of the Inverse Hessian in Large-Scale Variational Data Assimilation Problems

Alison Ramage

Department of Mathematics and Statistics, University of Strathclyde, Glasgow, Scotland
http://personal.strath.ac.uk/a.ramage/

Abstract

For a large-scale variational data assimilation problem, the Posterior Covariance Matrix (PCM) can be used to obtain important information, such as confidence intervals for the optimal solution. However, due to memory limitations, in practice it is often impossible to assemble, store or manipulate the PCM explicitly for a realistic model. One alternative approach is to approximate the PCM by the inverse Hessian of the auxiliary control problem based on the tangent linear model constraints. In many practical data assimilation problems, the majority of eigenvalues of the inverse Hessian are clustered in a narrow band above the unity, with only a relatively small percentage of the eigenvalues being distinct from this. A limited-memory representation of the inverse Hessian can therefore be built on the basis of a small (compared to the dimension of the state vector) number of ‘leading’ Ritz pairs of the projected Hessian, computed by the Lanczos method.

Although this approach is useful, taking into account the size of the state vectors used in modern realistic data assimilation applications (for example, $10^9$ to $10^{12}$ unknowns), using even a small percentage of the spectrum could still involve a significant number of eigenvalues, which may be far beyond the available storage capacity. In such cases, it is usually necessary to enhance the limited-memory algorithm by using an appropriate preconditioner. Within this framework, the standard application of a Control Variable Transform acts as a first level of preconditioning in the form of the square root of the background covariance matrix. However, if the impact of sensor information on the optimal solution is significant, this is not sufficient and an additional preconditioning step is required. In this talk we will present a limited-memory approximation to the inverse Hessian, computed using the Lanczos method, based on a multilevel preconditioning approach. We illustrate one use of this approximation by showing its potential effectiveness as a preconditioner within a Gauss-Newton iteration as part of a 4DVar simulation [1].

This work has been carried out in collaboration with Kirsty Brown (Strathclyde), Igor Gejadze (IRSTEA), Amos Lawless (Reading) and Nancy Nichols (Reading).

References

Randomized Discrete Empirical Interpolation Method for Nonlinear Model Reduction

Arvind K. Saibaba
Department of Mathematics, North Carolina State University
https://asaibab.math.ncsu.edu/

Abstract

Detailed mathematical models of physical phenomena, such as those used in weather prediction and neuroscience, require expensive simulation of large-scale models. The goal of model reduction is to replace computationally expensive full-scale models by reduced order models (ROMs) that are cheaper to evaluate and that approximate the full-scale models. We focus on a particular method for nonlinear model reduction called the Discrete Empirical Interpolation Method (DEIM) [1]. The DEIM interpolation framework computes an approximation of a nonlinear function \( f : \mathbb{R}^n \rightarrow \mathbb{R}^n \) using two ingredients: an orthonormal basis \( W \in \mathbb{R}^{n \times r} \) used to interpolate the function, and a set of indices, defining a point selection operator \( S \in \mathbb{R}^{n \times r} \), at which the nonlinear function is evaluated.

Both of these ingredients are expensive to compute when \( n \) is large. In recent work, randomized algorithms have been proposed for low-rank matrix approximations using random projections and random sampling; see the survey articles [2, 3]. Randomized methods have several advantages over their corresponding classical counterparts: typically, they are computationally efficient, numerically robust, easy to implement, and scale well in a distributed computing setting. Motivated by this success, we propose and develop new randomized algorithms for DEIM that enable its efficient large-scale implementation. We now briefly describe the major bottlenecks in computing the DEIM approximation and highlight our proposed modifications to lower the computational cost.

- The basis \( W \) is constructed as follows: several representative samples—also called as snapshots—of the function \( f(\cdot) \) are collected and arranged as columns of a matrix, known as the snapshot matrix. The left singular vectors of this snapshot matrix form the desired DEIM basis—henceforth, we call this the standard basis. The dimension of the subspace spanned by the DEIM basis, denoted by \( r \), depends on the number of dominant singular values of the snapshot matrix. Our approach replaces the SVD by a randomized SVD, lowering the cost from \( O(n^2) \) to \( O(rn_s) \), where \( n_s \) is the number of snapshots.

- Finding a set of good indices to define \( S \) is a combinatorially hard problem known as point selection in the DEIM literature (this is also related to subset selection). Various deterministic point selection techniques have been proposed in the literature that are based on the pivoted LU factorization, pivoted QR factorization, and strong rank-revealing QR factorization. We develop randomized point selection techniques which lower the cost of the deterministic algorithms, from \( O(nr^2) \) to \( O(nr) \) or \( O(nr + r^3 \log r) \) depending on the algorithm, and can exploit parallelism.

Contributions  As mentioned earlier, we develop randomized algorithms and analysis for tackling two major computational bottlenecks in large-scale implementation of DEIM. Below, we summarize the major contributions and highlight the novelty in our approach.
We present randomized approaches for efficiently constructing an approximate (orthonormal) DEIM basis $\hat{W} \in \mathbb{R}^{n \times r}$. The algorithms come in two flavors depending on whether the target rank $r$ is known in advance or not. When the target rank is known, we present algorithms based on the randomized subspace iteration. When the target rank $r$ is unknown, we provide an adaptive algorithm for computing $\hat{W}$. The computational and storage advantages of the randomized algorithms for constructing the DEIM basis will be highlighted.

We present a detailed analysis of the error when an approximate basis is used instead of the standard DEIM basis. A crucial component of our analysis involves the largest canonical angle between the subspaces spanned by the standard basis $W$, and an approximate basis $\hat{W}$. This analysis does not make additional assumptions regarding the approximate basis $\hat{W}$ and is, therefore, applicable beyond the context of randomized algorithms. Then, we develop bounds for the accuracy of the DEIM approximation tailored to the randomized algorithms listed above. These bounds use the analysis developed by the author in [4].

We propose two randomized point selection methods for DEIM approximation using randomized sampling: leverage score sampling and hybrid algorithms. For each method, we present theoretical bounds on the number of points required for the desired accuracy. The hybrid point selection technique combines the computational advantages of the randomized methods and the deterministic methods. Both these sampling methods have been developed in the context of matrix CUR decompositions; however, our analysis for the DEIM approximation is new.

Numerical experiments will demonstrate the computational benefits, the accuracy of the randomized approaches and insight into the choice of parameters for the algorithms presented.

We note that the contributions listed here are based on two papers [5, 4]. The second manuscript is currently under revision at SIAM Journal on Scientific Computing and a preprint has been posted to arXiv.

References


A Rational EVEN-IRA Algorithm for the solution of alternating polynomial Eigenvalue Problems

Heike Faßbender, Peter Benner, Philip Saltenberger

Institut Computational Mathematics, AG Numerik, TU Braunschweig
www.tu-braunschweig.de/icm/numerik/personal/phisalte

Abstract

In this contribution, a method for solving structured eigenvalue problems is presented. Eigenvalue problems are ubiquitous in engineering, physics, mechanics and many more scientific disciplines. Moreover, they lie at the heart of numerical linear algebra. As eigenproblems stemming from real-world-applications are often subject to physical constraints and side conditions, they frequently and naturally inherit structure. For instance, mechanical vibration systems are usually described by symmetric mass, damping and stiffness matrices, see Tisseur, Meerbergen [12]. Optimal control problems often involve Hamiltonian/skew-Hamiltonian matrix pencils (Mehrmann, Watkins [6]). But, after all, which features and properties single out faithful numerical algorithms for structured problems from universal methods? In the first place, the occurrence of structure can be utilized to speed up algorithms and reduce memory requirements. This originates from the deeper focus on the true nature of the problem compared to standard methods. In addition to that, the adequate exploitation of structure is beneficial (and indispensable, sometimes) for the reliability of an algorithm. Indeed, a proper numerical treatment of structure will often produce more accurate and physically meaningful results. Consequently, it seems reasonable to design tailor-made algorithms instead of addressing structured problems without any care by standard means.

We present an algorithm for real, alternating polynomial eigenvalue problems of large scale that takes into account all the aforementioned aspects. The method we propose is an implicitly-restarted rational Krylov-Schur approach based on the Even-Ira algorithm introduced by Mehrmann et al. in [5] (see also Faßbender, Saltenberger [4]). In contrast to the Even-Ira algorithm and motivated by the work of Benner, Effenberger [1], our approach explicitly allows for changes of the shift parameter during the iteration. This leads to a flexible and adjustable rational Krylov algorithm.

There exist various major applications from control theory leading to alternating polynomial eigenproblems of large size, see Mehrmann et al. [5], Betcke et al. [2]. A matrix polynomial

\[ P(\lambda) = \sum_{k=0}^{\ell} P_k \lambda^k = P_\ell \lambda^\ell + P_{\ell-1} \lambda^{\ell-1} + \cdots + P_1 \lambda + P_0 \in \mathbb{R}[\lambda]^{n \times n} \]

is called alternating if \( P_k = P_k^T \) holds whenever \( k \) is even and \( P_k = -P_k^T \) holds otherwise. Equivalently, \( P(-\lambda)^T = P(\lambda) \). Eigenvalue/eigenvector pairs \( (\mu, x) \in \mathbb{C} \times \mathbb{C}^n \) of \( P(\lambda) \) are characterized by the relation \( P(\mu)x = 0 \). To find eigenvalues of \( P(\lambda) \), it is a common approach to turn \( P(\lambda) \) into a matrix pencil \( L(\lambda) = \lambda A + B \) (e.g. the Frobenius companion form, see Mackey et al. [7]) by linearization. Then, the eigenvalues of \( P(\lambda) \) and \( L(\lambda) \) coincide and the generalized eigenproblem corresponding to the linearization \( L(\lambda) \) may be solved by, e.g., the standard QZ algorithm (see Moler, Stewart [8]). However, solving an alternating eigenvalue problem via the QZ algorithm and the Frobenius companion form is not conducive in the light of the problems nature and structure.

In particular, the spectral region of alternating matrix polynomials has a Hamiltonian structure, that is, the spectrum is symmetric with respect to both the real and the imaginary axis. Our
The algorithm takes care of this fact in two different ways. On the one hand, the linearization $L(\lambda) = \lambda A + B$ of $P(\lambda)$ we consider is an alternating matrix pencil itself (i.e. $B = B^T$, $A = -A^T$). Thus, it naturally preserves the Hamiltonian spectral structure of $P(\lambda)$. On the other hand, for any $\zeta \in \mathbb{C}$ outside the spectrum of $P(\lambda)$, we consider the special shift-and-invert transformation

$$L(\zeta) = \zeta A + B \mapsto K := L(\zeta)^{-T} AL(\zeta)^{-1} A$$

as proposed by Mehrmann, Watkins [6], Mehrmann et al. [5]. Each eigenvalue pair $(+\mu, -\mu)$ of $L(\lambda)$ is transformed to only one eigenvalue $\theta = (\mu^2 - \zeta^2)^{-1}$ of $K$. Consequently, $K$ preserves eigenvalue pairings and the spectral symmetry inherent to the problem is respected.

The foundation of our method is the EVEN-IRA algorithm from Mehrmann et al. [5] applied to $L(\lambda)$. This, in turn, is a sophisticated variant of the Krylov-Schur algorithm (see Stewart [11]) applied to $K$ for some appropriately chosen shift parameter $\zeta$. To define $K$, we take $L(\lambda) = \lambda A + B$ to be a special linearization from the class of SBMB pencils, see Dopico et al. [3]. Due to the structure and sparsity of $L(\lambda)$, the computation of matrix-vector-products $Kx$ can be realized implicitly without ever forming $K$ at all. Moreover, linear systems with $L(\zeta)$ and $L(\zeta)^T$ (that arise from matrix-vector-products with $K$) can be solved implicitly through systems involving only $P(\zeta)$ and $P(-\zeta)$. Accordingly, the complexity of computing $Kx$ is reduced by a significant amount since the size of $P(\zeta)$ is substantially smaller than the size of $L(\zeta)$. For the same reason, memory requirements (e.g. for storing matrix decompositions) can be decreased. As the EVEN-IRA algorithm does not allow for changes of the shift $\zeta$ during the iteration, this feature is incorporated in our method. Based on Benner, Effenberger [1] and Ruhe [10], our Rational EVEN-IRA algorithm permits shift adjustments during the iteration without discarding the information that has been accumulated so far. Retaining the advantageous computational aspects, this endows our approach with more flexibility. In consequence, the rational EVEN-IRA algorithm we present is a new reliable, flexible and fast numerical method with reasonable costs. To demonstrate its efficiency, we include numerical examples in the presentation.

References


Spectral estimates for saddle point matrices arising in numerical weather prediction

Ieva Daužickaitė, Amos S. Lawless, Jennifer A. Scott, and Peter Jan van Leeuwen

Department of Mathematics, University of Reading and
Scientific Computing Department, STFC Rutherford Appleton Laboratory
http://www.numerical.rl.ac.uk/people/j_scott/

Abstract

Data assimilation aims to compute a more accurate estimate of the state of a dynamical system by combining observations of the system with a prior estimate. The latter is termed the background state and is usually an output of a numerical model that is designed to simulate the dynamics of the system. The impact that the observations and the background state have on the state estimate depends on their errors, whose statistical properties are assumed to be known. Data assimilation is used to produce initial conditions in numerical weather prediction (NWP), and is also employed in a range of other areas, such as flood forecasting, research into atmospheric composition, and neuroscience. In operational applications, the process is made particularly challenging by the size of the system, e.g. the numerical model may be operating on in excess of $10^8$ state variables and $10^5 - 10^6$ observations may be incorporated. Moreover, there is usually a constraint on the time that can be spent on the computation and storage requirements present a further obstacle.

The sought-after solution is termed the analysis. One approach to computing the analysis is to solve a weighted least-squares problem, which involves minimising a cost function. An active research topic in this area is the weak constraint four-dimensional variational (4D-Var) data assimilation method (for example, [2, 4, 3]). It is employed in the search for states of the system over a time period, called the assimilation window. The cost function is formulated under the assumption that the numerical model is not perfect and it penalises the weighted discrepancy between the analysis and the observations, the analysis and the background state, and the difference between the analysis and the trajectory given by integrating the dynamical model.

The state of the dynamical system at times $t_0 < t_1 < \ldots < t_N$ is represented by the state vectors $x_0, x_1, \ldots, x_N$ with $x_i \in \mathbb{R}^n$. A nonlinear model $m_i$ describes the transition from the state at time $t_i$ to that at time $t_{i+1}$, i.e.

$$x_{i+1} = m_i(x_i) + \eta_{i+1}.$$

The model errors $\eta_i$ are assumed to be Gaussian with zero mean and covariance matrix $Q_i \in \mathbb{R}^{n \times n}$, and to be uncorrelated in time.

The background state at time $t_0$ is denoted by $x^b \in \mathbb{R}^n$. In NWP, $x^b$ usually comes from a previous short range forecast and is chosen to be the first guess of the state. It is assumed that the background state has errors that are Gaussian with zero mean and covariance matrix $B \in \mathbb{R}^{n \times n}$.

Observations $y_i \in \mathbb{R}^{p_i}$ ($p_i << n$) of the system at time $t_i$ are given by

$$y = \mathcal{H}_i(x_i) + \epsilon_i,$$

where the nonlinear observation operator $\mathcal{H}_i$ maps the state variables to the observation space. Approximations made when mapping the state variables to the observation space, different spatial and temporal scales between the model and some observations, and pre-processing, or quality control, of the observations comprise the representativity error. The observation error $\epsilon_i$ is then
made up of the representativity error combined with the error resulting from limited precision of the measurements. The observation errors are assumed to be Gaussian with zero mean and covariance matrix $R_i \in \mathbb{R}^{p_i \times p_i}$ and to be uncorrelated in time.

In weak constraint 4D-Var, the analysis $x_0^a, x_1^a, \ldots, x_N^a$ is computed by minimising the following nonlinear cost function

$$J(x_0, x_1, \ldots, x_N) = \frac{1}{2}(x_0 - x^b)^TB^{-1}(x_0 - x^b) + \frac{1}{2}\sum_{i=0}^{N}(y_i - \mathcal{H}_i(x_i))^TR_i^{-1}(y_i - \mathcal{H}_i(x_i))$$

$$+ \frac{1}{2}\sum_{i=0}^{N-1}(x_{i+1} - m_i(x_i))^TQ_{i+1}^{-1}(x_{i+1} - m_i(x_i)).$$

Effective minimisation techniques need evaluations of $J$ and its gradient that involve expensive operations with the dynamical model and its linearised variant. Such approaches are impractical in operational applications. One way to approximate the minimum of the weak constraint 4D-Var is to use a Gauss-Newton method that requires minimising a series of linearised quadratic cost functions and using the minima to update the state estimate. The state estimate update is found by solving large sparse symmetric linear systems using an iterative method.

In their work to increase the potential for using parallel computations when computing the state update with weak constraint 4D-Var, Fisher and Gürol [2] introduced a symmetric $3 \times 3$ block saddle point formulation. The resulting linear systems are solved using Krylov subspace solvers, whose convergence is affected by the spectra of the coefficient matrices. Previous attempts at developing robust practical preconditioners have proved largely ineffective [2, 4, 3] and an analysis of the spectra has been lacking. To address the latter, we derive bounds for the eigenvalues of the $3 \times 3$ block matrix $A_3$. We also reduce the system to a $2 \times 2$ block saddle point formulation and derive eigenvalue bounds for the system matrix $A_2$. Finally, we consider a $1 \times 1$ block formulation with a symmetric positive definite coefficient matrix $A_1$. Importantly, some of the blocks in the coefficient matrices depend on the available observations of the dynamical system. We present a novel examination of how adding new observations influences the spectra. Eigenvalue bounds and results on how the extreme eigenvalues and the bounds depend on the observations are presented. In particular, we show that the negative eigenvalues of $A_3$ and $A_2$ move away from the origin or are unchanged as observations are added but that the smallest positive eigenvalues can approach the origin, whilst the extreme eigenvalues of $A_1$ move away from the origin or are unchanged. Numerical experiments confirm the theoretical analysis. It is hoped that a better understanding of the spectra will help in the development of effective preconditioners. Our findings suggest that including information on the observations that comes from the observation error covariance matrix and the linearised observation operator could be beneficial; this is the subject of our ongoing research. Further details may be found in [1].

References


Graph-based PDEs: Laplacians, eigeninformation, and semi-supervised learning

Jessica Bosch, Steffen Klamt, Pedro Mercado, Martin Stoll

Department of Mathematics, TU Chemnitz
www.tu-chemnitz.de/mathematik/wire/

Abstract

Nonlinear PDE models are an ubiquitous tool in the modeling of real world phenomena. In particular, phase-field models such as the Allen–Cahn or Cahn–Hilliard equation have been employed to describe various problems from material science to image processing. In particular, these models can be used for repairing destroyed images by filling in the damaged regions while maintaining the undamaged image parts.

Recently, Bertozzi and Flenner [1] have defined a PDE based on the minimization of the energy

\[ E(u) = \frac{\varepsilon}{2} u^T L_s u + \frac{1}{4\varepsilon} \sum_{i=1}^{n} (u_i - 1)^2 + \sum_{i=1}^{n} \omega_i (f_i - u_i). \]

This energy is based on a graph \( G = (V, E) \) with \( V \) consisting of \( n \) nodes and a number of edges collected in \( E \). In the above function \( u_i \) is the phase-field function that should at convergence be at \(-1\) or \(1\) on all nodes of a graph, also \( f_i \) describes the training data, \( \varepsilon \) is the interface parameter and most importantly

\[ L_s = I - D^{-1/2}WD^{-1/2} \]

is the normalized graph Laplacian. Based on this well-studied matrix, we obtain the graph Allen-Cahn equation that is now used for the classification of the nodes and its efficiency depends on the efficient computation of the eigeninformation of \( L_s \). In many applications such as spatial data analysis or image processing the matrix \( L_s \) is large and dense rendering the eigenvalue computations infeasible. In this talk, we will demonstrate that Fourier-based methods can be used to compute the spectrum of \( L_s \) in a matrix-free way and illustrate that this can be used for spectral clustering or kernel ridge regression [2].

In several applications the data do not naturally come as a graph but are often naturally represented as a hypergraph. A hypergraph is defined by \( G = (V, E) \) where \( V \) again contains the \( n \) nodes that we want to classify. In contrast to classical graphs an edge in \( e \in E \) can now contain an arbitrary number of nodes. Also here the hypergraph can be represented by a graph Laplacian

\[ L_s = I - \Theta \]

with \( \Theta \) incorporating the adjacency terms and additional weight matrices. Note that this matrix is symmetric and positive semi-definite. We show that the eigeninformation of the hypergraph Laplacian can be used to speed up the solution of a differential equation for the purpose of semi-supervised learning [4]. We illustrate on several examples that the performance is competitive.

Additionally, we consider another type of graph that is important in social network analysis, namely signed networks. There the weight associated with an edge \( e \in E \) is either positive or negative modeling friend and foe relationships. In this case there are several competing graph Laplacians and we illustrate how these can be embedded into the framework of semi-supervised learning using the Allen–Cahn equation [3]. We again study how the eigeninformation can be used for the efficient classification of nodes in the social network.
References


Resolvent-based techniques for computing the discrete and continuous spectrum of differential operators

Matthew Colbrook, Andrew Horning, and Alex Townsend

Department of Mathematics, Cornell University
http://pi.math.cornell.edu/~ajt/

Abstract

Numerical methods for differential eigenproblems usually follow the “discretize-then-solve” paradigm. Discretize first, and then solve the matrix eigenproblem. The discretize-then-solve paradigm can be tricky for differential eigenproblems as the spectrum of matrix discretizations may not converge to the spectrum of the differential operator. Moreover, it is impossible to fully capture the continuous part of the spectrum with a finite-sized matrix eigenproblem. In this talk, we will discuss an alternative “solve-then-discretize” paradigm for differential eigenproblem. To compute the discrete spectrum, we will discuss a continuous analogue of FEAST [3] by approximating the action of the resolvent operator. For the continuous spectra, we will use a Cauchy-like integral to calculate a smoothed version of the spectral measure. Our algorithms for differential eigenproblems are adapted from linear algebra techniques for matrix eigenproblems.

FEAST for computing discrete part of the spectrum. Given an $n \times n$ matrix $A$ and a region in the complex plane $\Omega \subset \mathbb{C}$, a simple version of the FEAST matrix eigensolver computes the eigenvalues of $A$ in $\Omega$ in two steps.

(1) FEAST applies the spectral projector $P_\Omega$ to a suitable $n \times m$ matrix $Y$ (usually, $m \ll n$) to construct a basis for the eigenspace $V$ associated with the eigenvalues in $\Omega$. FEAST uses a Cauchy integral representation of $P_\Omega Y$, which is approximated by a quadrature rule. That is,

$$P_\Omega Y = \frac{1}{2\pi i} \int_{\partial \Omega} (zI - A)^{-1}Y dz \approx \frac{1}{2\pi i} \sum_{k=1}^\ell w_k (z_kI - A)^{-1}Y.$$  \hspace{1cm} (1)

(2) After extracting an orthonormal basis $Q$ from the columns of $P_\Omega Y$, the eigenvalues and eigenvectors of $A$ are approximated with the Ritz values and vectors of the $m \times m$ matrix $Q^*AQ$.

There are two primary advantages when applying this two-step strategy directly to a differential operator (i.e., replacing $A$ with an operator $L$): (1) One can derive error bounds to ensure there is no spectral pollution, and (2) Structured linear algebra can be exploited to guarantee accurate computations of high-frequency eigenfunctions. The formal algorithm (see Algorithm 1) must be discretized for a practical implementation; however, we avoid directly discretizing $L$.

**Algorithm 1** An operator analogue of FEAST for differential operators.

**Input:** $L : \mathcal{D}(L) \to \mathcal{H}, \Omega \subset \mathbb{C}$ containing $m$ eigenvalues of $L$, $F : \mathbb{C}^m \to \mathcal{H}$.

- Compute $V = P_\Omega F$.
- Compute $V = QR$, where $Q : \mathbb{C}^m \to \mathcal{D}(L) \subset \mathcal{H}$ has $\mathcal{H}$-orthonormal columns and $R \in \mathbb{C}^{m \times m}$ is upper triangular.
- Compute $L = Q^*LQ$ and solve $LX = \Lambda X$ for $\Lambda = \text{diag}[\lambda_1, \ldots, \lambda_m]$ and $X \in \mathbb{C}^{m \times m}$.

**Output:** Eigenvalues $\lambda_1, \ldots, \lambda_m$ in $\Omega$ and eigenfunctions $U = QX$. 

86
Cauchy-like integral for the continuous part of the spectrum. Given an $n \times n$ real, symmetric, and sparse matrix $A$, the weighted density of states (WDOS) with respect to a unit vector $v$ is given by [4]

$$\phi_v(\lambda) = \sum_{k=1}^{n} \langle P_k v, v \rangle \delta(\lambda - \lambda_k),$$

where $P_k$ is the spectral projector onto the eigenspace associated with the $k$th eigenvalue $\lambda_k$. An analogue for a self-adjoint differential operator $L$ is the so-called spectral measure:

$$\mu_f(\lambda) = \sum_k \langle P_k f, f \rangle_{\mathcal{H}} \delta(\lambda - \lambda_k) + \rho_f(\lambda), \quad \|f\|_{\mathcal{H}} = 1.$$  \hspace{1cm} (2)

Unlike the WDOS of a matrix, the spectral measure $\mu_f$ may have a density supported on the continuous spectrum of the differential operator (we assume $L$ has no singular continuous spectrum). Motivated by the FEAST eigensolver, we will discuss how one can compute a smoothed approximation to $\mu_f$ using a contour integral approach. That is,

$$\mu_f^\epsilon(\lambda) = \frac{1}{2\pi i} \int_{\Gamma} \frac{1}{1 + \lambda^2} \langle (L - z)^{-1} f, f \rangle_{\mathcal{H}} dz = \frac{1}{\pi} \text{Imag} \langle (L - (\lambda + i\epsilon))^{-1} f, f \rangle_{\mathcal{H}},$$

where $\Gamma$ is a carefully selected contour. Therefore, we will show how a smoothed approximation to $\mu_f$ can be computed by solving linear equations with complex shifts (see Algorithm 2). One can derive rigorous error bounds and convergence rates for the smoothed measure $\mu_f^\epsilon$ and leverage structured linear algebra for high-accuracy computation of $\mu_f$.

**Algorithm 2** Approximating the spectral measure of a differential operator.

**Input:** $L : \mathcal{D}(L) \rightarrow \mathcal{H}$, $\lambda_1, \ldots, \lambda_n \subset \mathbb{R}$, $f \in \mathcal{H}$ with $\|f\|_{\mathcal{H}} = 1$.  

- Solve $(L - (\lambda_k + i\epsilon))u_k = f$ for $u_k \in \mathcal{D}(L)$, $k = 1, \ldots, n$.  
- Compute $s = U^* f$, where $U$ is the quasimatrix $U = [u_1 \ldots u_n]$.

**Output:** Smoothed approximate spectral measure $s = [\mu_f^\epsilon(\lambda_1), \ldots, \mu_f^\epsilon(\lambda_n)]$.

**Application to stability analysis and scattering cross-sections.** Differential eigenvalue problems arise when analyzing the stability of a physical system subjected to small perturbations. Of particular interest and computational difficulty is the response of a system to high-frequency noise, which may model undesirable environmental interactions. We will show how one can exploit our FEAST method to assess the stability of an elastic beam subject to high-frequency perturbations. The study of how waves and particles scatter when interacting with an external potential usually leads to a differential eigenproblem in which the continuous part of the spectral measure $\mu_f$ plays a crucial role. We demonstrate how our algorithm for approximating $\mu_f$ can be used to study scattering cross-sections in a relativistic, quantum mechanical model and compare the results with those from the theory on resonant.

**References**


Learning from Network Data with Nonlinear Eigenvectors

Francesco Tudisco and Desmond J. Higham

School of Mathematics, Gran Sasso Science Institute, L’Aquila, Italy
https://ftudisco.gitlab.io

Abstract

An eigenvector-dependent nonlinear eigenvalue problem or, more briefly, a nonlinear eigenvector problem, has the form

\[ A(x)x = \lambda B(x)x, \]

where \( A, B : \mathbb{R}^n \to \mathbb{R}^{n \times n} \) are matrix valued functions. The linear (generalized) eigenvector problem is retrieved when both \( A \) and \( B \) are constant matrices. This type of eigenvector problem arises in many contexts, including quantum chemistry, physics, medical engineering, data science, image processing and machine learning.

In this talk I will present a number of results connecting nonlinear eigenvector problems to learning problems of central importance in network science and data mining that are related to the solution of constrained optimization problems of the form

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad g(x) = 1,
\end{align*}
\]

where \( f, g : \mathbb{R}^n \to \mathbb{R} \) are homogeneous, i.e., there exist real coefficients \( a, b \) such that \( f(\lambda x) = \lambda^a f(x) \) and \( g(\lambda x) = \lambda^b g(x) \) for all \( x \in \mathbb{R}^n \) and all \( \lambda > 0 \). For simplicity, I will also assume that both \( f \) and \( g \) are twice differentiable.

First, I will show that extremal nonlinear eigenvector solutions to (1) can be used to solve the constrained optimization problem (2).

**Theorem 1.** A vector \( x \) is solution of (2) if and only if there exist \( A \) and \( B \) such that \( x \) is a nonlinear eigenvector corresponding to the smallest eigenvalue solution of \( A(x)x = \lambda B(x)x \).

This straightforward result has many important consequences. For example, it allows us to approximate combinatorial quantities that frequently appear in machine learning and network science, and that are typically NP-hard. More precisely, assume we are given an undirected graph \( G = (V, E) \), with \( V = \{1, \ldots, n\} \), and consider the optimization problem

\[
\vartheta(G) = \min_{S \subseteq V} \frac{F(S)}{G(S)}
\]

where \( F, G : 2^V \to \mathbb{R} \) are set functions. This is a fundamental problem that appears very frequently in data mining and network science. For example, \( F \) can be the cut-function \( F(S) = \text{cut}(S) = |\{(i,j) \in E : i \in S, j \notin S\}| \) and \( G \) the balancing function \( G(S) = |S||V \setminus S| \). In this case, \( \vartheta(G) \) corresponds to the famous graph isoperimetric constant [1], which we denote by \( \gamma(G) \). While this quantity cannot be computed exactly, for large graphs the well-known Cheeger inequality shows that it can be approximated by the smallest nonzero eigenvalue \( \lambda^+ \) of the Laplacian matrix. Namely, we have \( \lambda^+ \leq \gamma(G) \leq O(1)\sqrt{\lambda^+} \).

Even though this inequality shows that \( \lambda^+ \) behaves like \( \gamma(G) \), the approximation obtained with the linear graph Laplacian can be far from sharp. Exploiting Theorem 1, we can improve the bounds by using suitable nonlinear eigenvectors. In fact, a result holds for any \( \vartheta(G) \) of the form (3).
Theorem 2. Let $F, G : 2^V \rightarrow \mathbb{R}$ be two nonnegative set functions such that $F(V) = G(V) = 0$. Then, for any $p > 1$ there exist $A, B : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ such that $\lambda_{A,B}^+ \leq \vartheta(G) \leq (\lambda_{A,B}^+)^{1/p}$, where $\lambda_{A,B}^+$ is the smallest nonzero eigenvalue solution of the nonlinear eigenvector problem $A(x)x = \lambda B(x)x$.

This theorem shows that suitable nonlinear eigenvector problems can significantly outperform standard eigenvector methods. However, in general, the former are more computationally challenging. This issue is clear from Theorem 2: when $p \rightarrow 1$ the approximation of $\vartheta(G)$ obtained by solving the nonlinear eigenvector problem in Theorem 2 can be arbitrarily good. However, computing $\vartheta(G)$ is known to be NP-hard (in general) and thus computing a prescribed solution to (1) must be NP-hard as well.

Moreover, Theorem 2 is based on the observation that one can transform (3) into (2) and the relation between (1) and (2). This provides further intuition on the computational challenge we have to face, as $f$ and $g$ are in general not convex and thus we cannot guarantee to compute a global solution to (2) in general.

The final result I will discuss addresses exactly this point. In particular, it shows that a global solution to (2) can be computed to an arbitrary precision for a large class of homogeneous functions $f$ and $g$, using the following nonlinear version of the power method

\[
\begin{align*}
    y_{k+1} &= A(x_k)x_k \\
    x_{k+1} &= y_{k+1}/\varphi(y_{k+1})
\end{align*}
\]

for suitable choices of $A : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ and $\varphi : \mathbb{R}^n \rightarrow \mathbb{R}_+$. In a way, this result is a nonlinear counterpart of the global convergence property of the standard power method for the solution of the nonconvex constrained optimization of the quadratic function $f(x) = x^T Ax$, subject to $g(x) = x^T x = 1$.

Theorem 3. Let $f$ and $g$ be homogeneous of degrees $a$ and $b$, respectively. Assume that the gradient of $g$ is invertible and that both $f$ and $g$ are positive, i.e. $f(x) > 0$ and $g(x) > 0$ for all $x > 0$ (entrywise). If $\kappa = |a - 1|/|b - 1| < 1$ then we can compute a global solution to (2) using a nonlinear power method of the form (4) that converges as $O(\kappa)$.

I will also discuss an analogous version of this theorem that involves (1) directly, rather than (2). As an illustration, I will apply this final result to the problem of core-periphery detection in networks [2].

I will also indicate how these results extend to the more general class of so-called multihomogeneous mappings [3].

References


Analytical Low-rank Compression via Proxy Point Selection

Jianlin Xia, Xin Ye, and Lexing Ying

Department of Mathematics, Purdue University
http://www.math.purdue.edu/~xiaj/

Abstract

In this talk, we study and analyze an innovative analytical low-rank compression method for some kernel matrices generated by a smooth kernel function $\kappa(x, y)$ evaluated at two well-separated sets of points $X = \{x_j\}_{j=1}^m$ and $Y = \{y_j\}_{j=1}^n$. Suppose $\kappa(x, y)$ has a degenerate expansion/approximation. Then the corresponding discretized kernel matrix as follows is numerically low rank:

$$K^{(X,Y)} \equiv (\kappa(x,y)_{x \in X, y \in Y}).$$

It has been known in potential theory that fast analytical low-rank approximation can be achieved for $K^{(X,Y)}$ via the use of proxy points. Suppose $Z = \{z_j\}_{j=1}^r$ is a set of proxy points located on an appropriate proxy surface enclosing the set $X$ (or $Y$), where $r$ is small. Then $K^{(X,Y)}$ has the following low-rank approximation:

$$K^{(X,Y)} \approx K^{(X,Z)} \Phi^{(Z,Y)},$$

(1)

where $\Phi^{(Z,Y)}$ is an appropriate matrix that can be conveniently obtained. (1) indicates that an approximate column basis matrix for $K^{(X,Y)}$ is simply $K^{(X,Z)}$. This basis matrix has a significant advantage of structure preservation. That is, it preserves the original kernel function information. This feature is very useful for fast structured algorithms such as those based on hierarchical matrices. Another benefit of the method is that it does not need to know the degenerate expansion for $\kappa(x, y)$.

This proxy point method thus gives a surprisingly convenient way of explicitly writing out approximate basis matrices for the kernel matrix. However, this elegant strategy is rarely known or used in the numerical linear algebra community. It still needs clear algebraic understanding of the theoretical background. Moreover, rigorous quantifications of the approximation errors and reliable criteria for the selection of the proxy points are still missing.

In this work, we present an intuitive way of understanding and analyzing the proxy point method for the low-rank approximation of some kernel matrices and give rigorous error analysis. We use contour integration to clearly justify the idea in terms of a class of important kernels. We further provide comprehensive accuracy analysis for the analytical compression and show how to choose nearly optimal proxy points. The main work includes:

- An intuitive explanation of this surprisingly simple analytical low-rank approximation method using contour integration so as to make this elegant method more accessible to the numerical linear algebra community.
- A hybrid analytical/algebraic low-rank compression strategy to get compact low-rank approximations and also to select certain representative points.
- Systematic analysis of the approximation errors of the proxy point method and the hybrid analytical/algebraic low-rank compression. For a given accuracy, we also show that $r$ is bounded and only depends on the separation of $X$ and $Y$ instead of their sizes.
Selection of a nearly optimal set of proxy points in the low-rank kernel matrix compression. Our error bounds give a clear guideline to control the low-rank approximation errors and to choose the locations of the proxy points so as to find nearly minimum errors.

This work thus gives a fast and reliable strategy for compressing those kernel matrices. Furthermore, it provides an intuitive way of understanding the proxy point method and bridges the gap between this useful analytical strategy and practical low-rank approximations. Some numerical examples help to further illustrate the ideas. Additional details are available in [1].

References

Communication-avoiding TT-tensor orthogonalization and rounding procedures

**Hussam Al Daas, Grey Ballard, and Peter Benner**

Department of Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems

https://www.mpi-magdeburg.mpg.de/person/97742/842836

Abstract

High dimensional data arise in various areas of scientific computing. Parametric PDEs, molecular simulations, and classification are examples among many others [5, 4, 6]. Even for a small number of modes, storing and manipulating explicit high dimensional data may become prohibitive, also known as the curse of dimensionality. For this reason, low-rank tensor approaches gained intensive attention of researchers in recent years. These methods allow to store the data implicitly and perform arithmetic operations on them with a reasonable complexity avoiding the curse of dimensionality. Most of the research focused on the establishment of representation formats and their corresponding arithmetic operations that reduce the floating-point operations complexity [5, 7, 3]. Only few papers have considered parallelizing tensor algorithms so far and mostly concentrated on compressing a tensor in full format into low-rank representation and performing tensor matrix contraction [2, 8, 1].

In this work, we present communication-avoiding algorithms for tensors represented in tensor train (TT) format. A $d$-dimensional tensor $X \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ is in TT format if

$$X(i_1, \ldots, i_d) = X_1(i_1) \cdots X_d(i_d), \forall i_k = 1, \ldots, n_k \text{ and } k = 1, \ldots, d,$$

where $X_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}$, $r_k \geq 1$, $k = 1, \ldots, d$, and $r_0 = r_d = 1$.

Left and right orthogonalization procedures play an important role in most computations with TT tensors, e.g., during the projection step in the Alternating Least Squares method, rounding of formal structures, etc. We analyze data distribution and communication cost of the orthogonalization and rounding procedures [1].

Using the latency-bandwidth model, our analysis shows that the asymptotic runtime estimation of the TT orthogonalization procedure is given as:

$$
\gamma \cdot \left( 5 \frac{d^3 n r^3}{P} + O(dr^3 \log P) \right) + \beta \cdot O(dr^2 \log P) + \alpha \cdot O(d \log P),
$$

(1)

where $P$ is the number of processors, $r_k = r$, $k = 1, \ldots, d - 1$, and $n_k = n$, $k = 1, \ldots, d$. The parameters $\alpha, \beta,$ and $\gamma$ are the latency, the inverse bandwidth, and the inverse floating-point rate, respectively.

Due to the sequential scheme of TT tensors with respect to the modes, the performance of parallel orthogonalization and rounding algorithms becomes quickly communication-bound for a high number of modes, $d$; see the latency cost in (1). To tackle this issue, we use a mixed representation TT-Tucker as pointed out in the literature [7]. We introduce a communication-avoiding orthogonalization and rounding procedures for TT-Tucker representation. Using the same previous notation, we show that the asymptotic runtime estimation of the TT-Tucker orthogonalization procedure is:

$$
\gamma \cdot \left( 4 \frac{d^3 n t^2}{P} + O \left( dt^3 \log \frac{P}{d} \right) + O(dr^2 t^2) + O(dr^3 t) \right) + \beta \cdot O \left( t^2 \log \frac{P}{d} \right) + \alpha \cdot O \left( \log \frac{P}{d} + d \right),
$$

(2)
where $t$ is the Tucker rank (supposing $t_k = t$ for $k = 1, \ldots, d$) of the tensor. Hence, a factor of $d$ in terms of communication cost is saved with respect to the TT variant.

Furthermore, by exploiting the implicit representation of the Q factor in the communication-avoiding QR, we can reduce the arithmetic complexity by a factor 2 for TT and TT-Tucker orthogonalization procedures.

Numerical experiments on large numbers of processors demonstrate the scalability of the proposed methods.

References


A Localization Preserving Preconditioner for Nonlinear Eigenvector Problems under Disorder

Robert Altmann and Daniel Peterseim
Computational Mathematics, University of Augsburg
https://www.uni-augsburg.de/de/fakultaet/mntf/math/prof/numa/team/robert-altmann/

Abstract

Localization is a universal wave phenomenon which occurs under a sufficiently large degree of disorder [3]. For the linear Schrödinger eigenvalue problem $-\Delta u + Vu = \lambda u$ this means that the eigenfunctions of low energy essentially localize in the sense of an exponential decay of their moduli if the potential $V$ is strongly oscillatory with high amplitudes.

This astonishing effect carries over to the more involved and nonlinear Gross-Pitaevskii eigenvalue problem, which has the form

$$-\Delta u + Vu + \delta |u|^2 u = \lambda u.$$ (1)

More precisely, low-energy states are still of localized nature if the parameter $\delta \geq 0$ is of moderate size, depending on the spatial dimension [2]. In physics, this parameter models the strength of repulsive particle interactions, which form a so-called Bose–Einstein condensate.

A spatial discretization of (1) leads to a large nonlinear eigenvector problem, i.e., a nonlinear eigenvalue problem with the nonlinearity in the eigenvector. Here, the large system dimension comes from the fact that a very small mesh size is necessary in order to resolve the oscillatory behavior of the potential. For the linear problem, i.e., for $\delta = 0$, we have constructed a reliable numerical scheme, which provides localized approximations of localized eigenvectors [1]. The algorithm is based on a preconditioned inverse iteration including a multigrid solver, which spreads information only locally. In this talk we present a generalization of this idea to the nonlinear setup with $\delta > 0$. For this, we combine localization preserving preconditioning with a globally convergent iteration scheme for the nonlinear eigenvector problem. The ability of the algorithm to transfer all computations to a local level then enables fast and reliable approximations of the low-energy eigenpairs.

References


Reduction of Systems with Polynomial Nonlinearities in the Loewner Framework

Athanasios C. Antoulas, I. V. Gosea, and M. Heinkenschloss

Department of Electrical and Computer Engineering, Rice University, Houston, USA
DRI Group, Max Planck Institute, Magdeburg, Germany
Baylor College of Medicine, Houston, USA

Abstract

Model order reduction (MOR) is a methodology commonly used to transform large-order models of time-dependent processes into much smaller models that are still able to represent the characteristics of the original process under a variety of conditions.

In this contribution, the main focus will be directed to the application of MOR for special classes of nonlinear systems, i.e. polynomial systems.

Systems with polynomial nonlinearities arise in many applications, mostly with cubic and quadratic terms. These include systems characterized by heat transfer with temperature dependent diffusivity, the Navier-Stokes or Burgers’ equations, the Chaffee-Infante equations, and also in modeling activation and deactivation dynamics of a spiking neuron, i.e. the FitzHugh-Nagumo system.

The main MOR method discussed here is the Loewner framework. It is a data-driven interpolation based approach that computes a reduced-oder model (ROM) whose transfer function (approximately) interpolates the transfer function of the original system. An attractive feature of the Loewner framework is that the ROM is computed using only transfer function measurements and does not require explicit access to the system matrices. For an exhaustive tutorial paper, see [4]. Recently, the framework has been extended to bilinear systems in [3], and to quadratic-bilinear (QB) systems in [7] and [1]. Other methods for MOR of QB systems have also been proposed in the literature, e.g. two-sided projection-based in [5] or quasi-optimal iterative-based in [6].

In this contribution, we study the impact of different lifting transformations, which transform a system with polynomial nonlinearity into a QB system, on the computation of reduced order models (ROMs) in the Loewner framework. More details are given in [2].

We analyze large-scale polynomial systems characterized by the following equations

$$\begin{align*}
E \frac{d}{dt} x(t) &= A_1 x(t) + \sum_{k=2}^{K} A_k x(t) \otimes \ldots \otimes x(t) + B u(t), \\
y(t) &= C_1 x(t) + \sum_{k=2}^{K} C_k x(t) \otimes \ldots \otimes x(t) + D u(t),
\end{align*}$$

(1)

with polynomial nonlinearities in the state and (possibly) in the output equation. The dimensions of the system matrices in (1) are as follows (for $k = 1, \ldots, K$ with $K, m, n, p$ being positive integers)

$$E \in \mathbb{R}^{n \times n}, \quad A_k \in \mathbb{R}^{n \times n^k}, \quad B \in \mathbb{R}^{n \times m}, \quad C_k \in \mathbb{R}^{p \times n^k}, \quad D \in \mathbb{R}^{p \times m}. \quad (2)$$

Instead of applying MOR methods directly to the system introduced in (1), we first reformulate the system and equivalently lift it to a QB system. This is performed by using specifically tailored lifting transformations. More specifically, auxiliary variables and equations are introduced in order
to recast the equations into a desired quadratic-bilinear structure. This allows the application of conventional MOR methods to more general nonlinear systems. Specific lifting transformations have already been discussed in [5], [8], and [9].

First, put together an augmented internal variable vector $x$ that contains powers of the original internal variable vector $x$; more specifically Kronecker products of $x$ with itself. Let $x^{(k)} \in \mathbb{R}^n$ denote the product $x \otimes \cdots \otimes x$ that contains $k$ instances of $x$. Then, introduce $x = ((x^{(1)})^T, \ldots, (x^{(K)})^T)^T \in \mathbb{R}^L$ as the internal variable corresponding to the lifted system (3), where $L = \sum_{k=1}^K n^k$. By explicitly writing the derivative of the $x^{(k)}$ terms from $x$, grouping terms together and rearranging, we obtain the polynomial system in (1) as a QB system

$$E \frac{d}{dt} x(t) = Ax(t) + Q(x(t) \otimes x(t)) + N(u(t) \otimes x(t)) + Bu(t), \quad y(t) = Cx(t) + Du(t). \quad (3)$$

The dimensions of the system matrices in (3) are as follows

$$E \in \mathbb{R}^{L \times L}, \quad A \in \mathbb{R}^{L \times L}, \quad B \in \mathbb{R}^{L \times m}, \quad C \in \mathbb{R}^{p \times L}, \quad D \in \mathbb{R}^{p \times m}, \quad Q \in \mathbb{R}^{L \times L^2}, \quad N \in \mathbb{R}^{L \times Lm}. \quad (4)$$

Clearly, the explicit computation of the matrices $Q$ and $N$ is tedious and unfeasible even for moderate dimensions of $n$. That is why we will not compute these matrices explicitly, but only the output of the bilinear maps $(x, x) \rightarrow Q(x \otimes x)$ and $(x, u) \rightarrow N(u \otimes x)$, i.e. a vector of length $L$.

One resulting aspect is that, in general, systems with polynomial nonlinearities can be transformed exactly into QB systems using lifting transformations. However, the lifting transformation is not unique. Actually one can build infinitely many such transformations by introducing free parameters. Additionally, different lifting transformations lead to different QB systems. While the QB systems are equivalent to the original system with polynomial nonlinearity, MOR approaches applied to the different QB systems may generate different ROMs and the approximation quality could drastically differ from case to case.

The purpose of this contribution is (1) to present different lifting formulations and their impact on the transfer functions associated with the resulting QB system, and (2) to provide guidelines for selecting lifting transformations in an automatic way (this may depend on the application). We study the ROMs computed using the Loewner framework applied to the QB systems that are obtained by applying several classes of transformations. The different approximation properties of the resulting ROMs are traced back to the different transfer functions associated with the different QB formulations.

The choice of lifting transformations could also be important for other approaches that are based on matching certain input-output maps of the system (double-sided moment-matching, iterative IRKA-type, etc.).

Finally, we present numerical simulation examples that include the FitzHugh-Nagumo system and the Chaffee-Infante equations.

References


Interpolation spaces: a numerical linear algebra perspective

Mario Arioli,
Universit` a LUM Jean Monnet, Italy
https://www.researchgate.net/profile/Mario_Arioli
mario.arioli@gmail.com

Abstract

Let $M \in \mathbb{R}^{m \times m}$ and $N \in \mathbb{R}^{n \times n}$ be symmetric positive definite matrices. By using $M$ and $N$, we define the scalar products $(x, y)_M = x^T M y$ and $(x, y)_N = x^T N y$, and then the following Hilbert spaces

$M = \{ \in \mathbb{R}^m; \| x \|^2_M = T^T M \}$, \quad $N = \{ q \in \mathbb{R}^n; \| q \|^2_N = q^T N q \}$

and their dual spaces

$M^* = \{ w \in \mathbb{R}^m; \| w \|^2_{M^-1} = w^T M^{-1} w \}$, \quad $N^* = \{ y \in \mathbb{R}^n; \| y \|^2_{N^-1} = y^T N^{-1} y \}$.

Introducing the matrix $S = M^{-1} N$ we have that $S$ is self-adjoint in the scalar product defined by $M$:

$(u, )_N = (u, S)_M = (Su, )_M$.

Furthermore, the eigenvalues of $S$ are the generalised eigenvalues of

$N x = \mu M x$,

i.e.

$\exists W \text{ s.t. } M = W^T W, \quad N = W^T \Delta^2 W, \quad \Delta > 0$ diagonal.

Thus, we have $S = W^{-1} \Delta^2 W$ and $M S^\theta = W^T \Delta^{2\theta} W$. We introduce finite dimensional interpolation spaces as follows:

$[N, M]_\theta = \{ u \in \mathbb{R}^n; \left( (u, S^{1-\theta} u)_M \right)^{1/2} \}$.

These interpolation spaces are also finite dimensional Hilbert spaces and the symmetric positive definite matrices defining the scalar products are:

$H_\theta = MS^{1-\theta} = W^T \Delta^{2(1-\theta)} W$.

Moreover, taking into account the properties of the dual interpolation spaces [5, 6], we have that

$[N, M]^\star = [M^\star, N^\star]_{1-\theta}$

and the matrix of our finite dimensional dual spaces $H_{1-\theta}$ are

$H_{1-\theta}^\star = N^{-1} (NM^{-1})^\theta = W^{-1} \Delta^{2(\theta-1)} W^{-T} = H_\theta^{-1}$.

In [1], we proved, using the Interpolation Theorem [5, Ch.1], that the finite dimensional interpolation spaces approximate the continuous ones when finite element methods are used to discretize the classical operator defining the continuous Sobolev’s spaces. Moreover, in [1], we gave a method based on Lanczos’ algorithm able to approximate either $H_\theta u$ or $H^{-1}_\theta f$. Here, we take advantage of
the K-Method theory [6] and propose a novel algorithm for the computation of the above problems. The algorithm is based on the following relations [5, Theorem 15.1, pp. 108–109]:

\[ c_{H_\theta} u = \left( \int_0^{+\infty} t^{1-2\theta} N(M + t^2 N)^{-1} M dt \right) u, \]

\[ c_{H_\theta^{-1}} f = \left( \int_0^{+\infty} t^{1-2\theta} N^{-1} (M^{-1} + t^2 N^{-1})^{-1} M^{-1} dt \right) f = \left( \int_0^{+\infty} t^{1-2\theta} (M + t^2 N)^{-1} dt \right) f, \]

\[ (c = \int_0^{+\infty} \frac{\mu^{1-2\theta}}{1 + \mu^2} d\mu), \quad 0 < \theta < 1. \]

Both the above integrals can be approximated (with high accuracy) by quadrature formulas

\[ c_{H_\theta} u = \sum_{k=1}^{N} \omega_k t_k^{-2\theta-1} N(M + t_k^2 N)^{-1} M u, \]

\[ c_{H_\theta^{-1}} f = \sum_{k=1}^{N} \omega_k t_k^{-2\theta-1} (M + t_k^2 N)^{-1} f \]

where the points \( t_k \) and the weights \( \omega_k \) can be chosen following either a Gaussian quadrature or a Fourier-Chebyshev quadrature [2]. The advantage of this approach is the high parallelism intrinsic to the method (all the systems corresponding to each \( t_k \) are independent) and, if \( M \) and \( N \) are sparse, we can take advantage of sparsity, while the matrices \( H_\theta \) are always dense.

Finally, we will show how the above results can be used to solve fractional derivatives or non-local operators problems related to statistics [7] applied to physical problems [3, 4] and economics.

References


Randomization for the Efficient computation of Reduced Order Models

Selin Aslan, Eric de Sturler and Serkan Gugercin

X-ray Science Division, Argonne National Laboratory
https://iola.math.vt.edu/people/selin/

Abstract
Nonlinear inverse problems appear in many applications for identification and localization of anomalous regions, such as finding tumors in the body, luggage screening, and finding contaminant pools in the earth. In this work, we focus on diffuse optical tomography (DOT) in medical image reconstruction; however, our methods have the potential to be useful for other applications as well. In DOT, we aim to recover an unknown image of interest, such as cancerous tissue in a given medium, using a mathematical model (the forward model) combined with measurements. The forward model in DOT is a diffusion model for the photon flux. The main computational bottleneck in such inverse problems is the repeated evaluation of a large-scale forward model. For DOT, this corresponds to solving large linear systems for each source and frequency at each optimization step. In addition, as Newton methods are very effective for these problems, we need to solve linear systems with the adjoint for each detector and frequency at each optimization step to efficiently compute derivative information. As rapid advances in technology allow for large numbers of sources and detectors, these problems become computationally prohibitively expensive. Hence, we need new computational techniques that characterize the medium quickly and efficiently for such inverse problems.

In the past, the use of reduced order models (ROM) has been proposed to drastically reduce the size of the linear systems solved in each optimization step in DOT, while still solving the inverse problem accurately [1]. This approach significantly reduces the cost of the inversion process by drastically reducing the computational cost of solving the forward problems. However, interpolatory model reduction requires the solution of large linear systems for all sources and frequencies as well as for all detectors and frequencies for each interpolation point in parameter space, followed by an expensive rank-revealing factorization to reduce the dimension. Hence, as the number of sources and detectors increases, even the construction of the ROM bases still incurs a substantial cost in the offline stage.

Another approach to drastically reduce the number of vectors in a (nearly) low rank matrix is to multiply the matrix by a modest number of random vectors to get the low rank structure via sampling [2]. In our case, first generating the bases and then sampling is very expensive. Instead in [3], we implement this approach using a few random linear combinations of the right-hand sides as stochastic sources and detectors and solve only for this modest number of resulting right hand sides in order to improve convergence for DOT.

To give some perspective on the problem, consider a 3D test problem using $32 \times 32 \times 32$ mesh with 225 sources and detectors with 3 parameter sample points and 4 frequency interpolation points. Using the full order model (FOM), the optimization algorithm solves 47,700 linear systems of dimension 32,768 to reconstruct the absorption image. The standard method for computing the global ROM basis with all sources and detectors [1] needs to solve 5400 large linear systems of dimension 32,768. However, after a rank-revealing factorization only 1228 of those directions are used for the global ROM basis. This reveals that the standard method for ROM construction solves many more linear systems than needed.
Since candidate ROM bases are (nearly) low rank, we propose to employ randomization to capture essentially the same subspace at much lower cost. Recall that using ROMs drastically reduces the size of the linear systems solved in each step of the optimization [1] and our randomization technique [3] reduces the number of large linear system solves needed for the ROM basis, combining these two approaches results in an effective and computationally highly efficient approach for nonlinear parameter inversion. Furthermore, using randomization for the efficient computation of ROM can be applied to numerous large-scale optimization problems.

To present a proof-of-concept experiment for the test problem above, constructing ROM with stochastic sources and detectors only requires 1,200 linear systems of dimension 32,768. Hence, our approach reduces the large solver cost by about a factor 40, while obtaining similar quality reconstruction results. For the main application discussed in this work, DOT, the size of a realistic linear system is at least $O(10^6)$. The number of sources and detectors may be a thousand or more, and combined with multiple frequencies, the resulting computational cost is indeed very high. Using randomization, we propose to drastically reduce the number of large linear solves for computing the global ROM basis for DOT. Although we focus on DOT, the ideas presented in this talk are relevant to many other large scale applications as well.

Furthermore, we plan to provide a brief theoretical justification for exploiting low rank structure in the reduction basis. Then, we link our approach, using randomization to compute the interpolatory model reduction bases, to tangential interpolation.

References


Meeting the challenges of computing many eigenpairs

Zhaojun Bai, Jack Dongarra, Chao-Ping Lin, Ding Lu and Ichitaro Yamazaki

Department of Computer Science and Department of Mathematics
University of California, Davis

www.cs.ucdavis.edu/~bai

Abstract

Continuing challenges of eigenvalue problems come from increasingly large problem sizes and needs of rapid and robust solvers due to the fact that eigenvalue problems are often in the inner-loop (kernel) of scientific computing problems and can be the bottlenecks of simulation efficiency. Over the years, we have seen that the computation of small numbers of eigenpairs are no longer able to meet the demands of emerging applications. These emerging applications include the investigation and design of new materials such as Lithium-ion electrolyte and graphene and the study of dynamics analysis of viral capsids of supramolecular systems such as Zika and West Nile viruses. These applications actually need large number of eigenpairs of large matrices at the order of thousands or more.

We have been involved in synergistic efforts to improve numerical methods for large eigenvalue problems, where the word large has two meanings: large in dimensionality (degrees of freedom) and large in the number of the required eigenpairs. In fact, the latter “large” poses significant challenges to existing algorithms and software. These algorithms and solvers have not been designed specifically for handling the situation when the number of targeted eigenpairs is large and when the eigenvalues are located well inside the spectrum.

Large eigenvalue problems pose challenges due to the fact that algorithmically, it becomes increasingly difficult to find interior eigenvalues, and computationally, it progressively requires more memory and data communication. These challenges are further escalated on computing platforms such as high-end desktop computers and multi-core nodes with shared memory that most computational scientists and engineers are working on. We believe that the capability of being able to efficiently compute many eigenpairs will not just be appealing, but also mandatory for the next generation of eigensolvers. We will discuss two core techniques to improve existing Krylov subspace methods: (1) Explicit external deflation (EED) for moving away the computed eigenpairs to prevent the algorithm from computing over again those quantities, and (2) Communication-avoiding matrix powers kernel (MPK) for fast matrix-vector products in Krylov subspace solvers with the EED.

The EED is a classical technique dated back to Hotelling in 1930s. In a slightly generalized formulation for a symmetric matrix $A$, the EED is based on the Hotelling’s theorem that if $U_k$ consists of eigenvectors of $A$ corresponding to the $k$ eigenvalues $\lambda_1(A), \ldots, \lambda_k(A)$, then the eigenvalues of $A_k = A + \sigma U_k U_k^T$ are $\lambda_j(A_k) = \lambda_j(A) + \sigma$ for $j \in \{1, \ldots, k\}$ and $\lambda_j(A_k) = \lambda_j(A)$ for $j \in \{k + 1, \ldots, n\}$. The eigenvectors of $A$ and $A_k$ stay unchanged. By Hotelling’s theorem, we immediately see that once a batch of eigenpairs is computed, one can pick a proper shift $\sigma$ to move them away, and then compute the next batch of eigenpairs of $A$ using the matrix $A_k$. The EED has a number of desired properties for large problems such as providing a memory-saving option over internal locking, which is a standard implementation in ARPACK, TRILAN and other packages, and easy extension to treat the generalized symmetric eigenproblem $Ax = \lambda Bx$ through the low-rank updated problem $(A + \sigma BU_k U_k^T B)x = \lambda Bx$ without the need of factoring $B$. This is in striking contrast to those algorithms where one has to involve the approximation of the actions of $B^{-1}$ on
vectors in order to avoid the direct factorization of $B$. We will present our recent analysis on the stability of the EED.

The MPK is a technique to reduce the computation and communication costs for generating the Krylov basis vectors: $[p_0(A_k)v, p_1(A_k)v, \ldots, p_s(A_k)v]$ with the sparse plus low-rank matrix $A_k = A + \sigma U_k U_k^T$, where $p_j(\cdot)$ are polynomials defined recursively, $\sigma$ is a shift and $U_k$ are the eigenvectors corresponding to the eigenvalues $\Lambda_k = \text{diag}(\lambda_1, \ldots, \lambda_k)$. In the spirit of reducing communication on the dense eigenvectors $U_k$ for computing the MPK, we have applied a blocking covers scheme designed for out-of-core algorithms for linear relaxation by Leiserson, Rao and Toledo in 1997 and an algorithm for exploiting data sparsity in parallel matrix powers computations by Knight, Carson and Demmel in 2013, to derive a communication-avoiding algorithm for computing the MPK (CA-MPK). We will show that CA-MPK algorithm accesses $U_k$ only twice while the standard algorithm need to access $U_k$ 2s times within the for-loop. In addition to reducing the communication cost, the CA-MPK algorithm takes advantages of the eigenvector property of $U_k$ and reduces the computational cost.

We will report the latest status of software development and applications.
Block Modified Gram-Schmidt Algorithms and their Analysis

Jesse L. Barlow

Computer Science and Engineering, The Pennsylvania State University
University Park, PA 16802 USA
www.cse.psu.edu/~barlow

Abstract

For given integers \( m \) and \( n \), \( m \geq n \), and a full column rank matrix \( X \in \mathbb{R}^{m \times n} \), we develop block modified Gram-Schmidt (BMGS) algorithms that factor \( X \) into \( Q \in \mathbb{R}^{m \times n} \) and upper triangular, nonsingular \( R \in \mathbb{R}^{m \times n} \) such that

\[
X = QR
\]

where, in exact arithmetic, \( Q \) is left orthogonal, i.e., \( Q^TQ = I_n \).

Our BMGS algorithms are built upon rewriting the modified Gram-Schmidt (MGS) Q-R factorization as a BLAS-3 algorithm, that is, one primarily based upon matrix-matrix operations, thus making it more efficient on cache-based computer architectures and more suitable for some distributed computing environments.

The framework for BMGS blends two closely related ideas. The first is a connection between Householder and MGS Q-R factorization first observed by Charles Sheffield and communicated to Gene Golub. That observation states that the matrices \( Q \) and \( R \) from MGS implicitly produce the Q-R factorization

\[
X \overset{\text{def}}{=} \begin{pmatrix} 0_{n \times n} & X \end{pmatrix} = U \begin{pmatrix} R \\ 0_{m \times n} \end{pmatrix}
\]

where \( U \in \mathbb{R}^{(m+n) \times (m+n)} \) is product of Householder matrices. Moreover, this equivalence holds in finite precision arithmetic. We design BMGS algorithms that are able to maintain this structure with small backward error in floating point arithmetic.

The Sheffield structure has been used by Björck and Paige [3] and by Paige, Rozložník, and Strakoš [6] in the development and analysis of MGS related algorithms. It has also been used by Barlow, Bosner, and Drmač [2], by Bosner and Barlow [4], and by Barlow [1] in the development of bidiagonalization algorithms. Paige [5] gives an exposition of the theory behind Sheffield’s observation.

The second idea is the Schreiber-Van Loan [7] representation of products of Householder transformations. Since the columnwise version of MGS is a BLAS-1 algorithm, this representation is useful in developing BLAS-2 and BLAS-3 versions of MGS. We construct a BLAS-3 version that uses a “tall, skinny Q-R” (TSQR) factorizations other than MGS for a key intermediate computation. This structure allows us to give sufficient conditions for a BMGS algorithm to have similar properties in floating point arithmetic to those of MGS. Previous BMGS algorithms can also be understood in the framework presented here.

Gram-Schmidt based algorithms are important for the implementation of Krylov space methods such as GMRES [6]. Block GMRES methods for linear systems with multiple right hand sides are discussed by Soodhalter [9] and Simoncini and Gallopoulou [8].
References


Reduction of Time-Periodic Systems: the Floquet-Arnoldi Decomposition

Christopher Beattie

Department of Mathematics, Virginia Tech
https://www.math.vt.edu/people/faculty/beattie-christopher.html

Abstract

Time-periodic systems are ubiquitous both naturally and as engineered systems. This is often as a consequence of periodic forcing due to rotation (e.g., the Earth’s rotation generates both tidal gravity forces and diurnal temperature gradients that cyclically drive atmospheric and ocean flows; gyroscopic forces can generate significant vibration and noise in vehicles). but more broadly, periodic phenomena can occur through the emergence of a dynamic balance between inertial and various restoring forces. For example, a structure exposed to an otherwise steady wind- or current-periodic phenomena can occur through the emergence of a dynamic balance between inertial and tidal gravity forces and diurnal temperature gradients that cyclically drive atmospheric and ocean as a consequence of periodic forcing due to rotation (e.g., the Earth’s rotation generates both

Time-periodic systems are ubiquitous both naturally and as engineered systems. This is often as a consequence of periodic forcing due to rotation (e.g., the Earth’s rotation generates both tidal gravity forces and diurnal temperature gradients that cyclically drive atmospheric and ocean flows; gyroscopic forces can generate significant vibration and noise in vehicles). but more broadly, periodic phenomena can occur through the emergence of a dynamic balance between inertial and various restoring forces. For example, a structure exposed to an otherwise steady wind- or current-periodic phenomena can occur through the emergence of a dynamic balance between inertial and tidal gravity forces and diurnal temperature gradients that cyclically drive atmospheric and ocean as a consequence of periodic forcing due to rotation (e.g., the Earth’s rotation generates both

Linear time-periodic (LTP) systems play a fundamental role in the analysis, simulation, and control of such phenomena even when the underlying models reflect fundamentally nonlinear dynamics, since by their character the periodic phenomena of interest emerge as components of a “center manifold” and must themselves be stable at least when subjected to small perturbations. Were this not the case, say for a natural system, oscillatory phenomena would not generally be observed, while for an engineered system, they would not generally be desired. Beyond this, computational strategies for extracting periodic solutions of nonlinear systems necessitate repeated solution of linear periodic systems, and this leads (naturally) to the the question of effective model order reduction for such systems.

We consider large-scale time-periodic dynamical systems of the form:

\[
E(t)\dot{x}(t) = A(t)x(t) + B(t)u(t) \\
y(t) = C(t)x(t) \tag{1}
\]

where \(E(t), A(t) \in \mathbb{R}^{n \times n}\) and \(B(t) \in \mathbb{R}^{n \times p}\), and \(C(t) \in \mathbb{R}^{m \times n}\) are \(T\)-periodic, with \(E(t) = E(t+T), A(t) = A(t+T), B(t) = B(t+T)\) and \(C(t) = C(t+T)\) for some fixed \(T > 0\).

For any given order \(r \ll n\), our goal is to find a reduced-order time-periodic linear model,

\[
E_r(t)\dot{x}_r(t) = A_r(t)x_r(t) + B_r(t)u(t), \quad y_r(t) = C_r(t)x(t) \tag{2}
\]

where \(E_r(t), A_r(t) \in \mathbb{R}^{r \times r}\) and \(B_r(t) \in \mathbb{R}^{r \times n_i}\), and \(C_r(t) \in \mathbb{R}^{r \times n_o}\) are obtained via projection so as to be also \(T\)-periodic, and chosen in such a way so that \(y_r(t) \approx y(t)\) over a wide class of admissible inputs \(u(t)\). Given an LTP system as in (1), the Floquet transformation is defined via the monodromy matrix, the fundamental solution matrix for the homogeneous system evaluated at \(T\) (the system period): Given \(\Phi(t)\) such that \(\Phi(0) = I\) and \(E(t)\Phi = A(t)\Phi\), the monodromy matrix is \(\Phi(T)\). With a suitable choice of branch cut for the (complex) logarithm, one may introduce \(R_0 = \frac{1}{i}\log \Phi(T)\). It then follows that \(\Phi(t) = \Phi(t)e^{-(R_0 t)}\) is \(T\)-periodic and this leads to a time-periodic change of variable \(\mathbf{z}(t) = \mathbf{P}^{-1}(t)x(t)\) such that

\[
E(t)\dot{\mathbf{z}}(t) = A(t)x(t) + B(t)u(t) \\
y(t) = C(t)x(t) \quad \Rightarrow \quad \begin{cases} \dot{\mathbf{z}}(t) = R_0 \mathbf{z}(t) + \mathbf{P}^{-1}(t) \mathbf{B}(t)u(t) \\
y(t) = \mathbf{C}(t) \mathbf{z}(t) \end{cases} \tag{3}
\]
A key consequence of this transformation is that the time-dependence in the system has now been isolated in the input/output ports. Our interest in this feature is that this allows one to take advantage of powerful model reduction methods that originally were designed for linear time-invariant systems. Although Floquet transformation is not normally viewed as a computational tool, for problems of small to moderate size, algorithms have been developed over the past decade that make effective use of Fourier spectral methods to identify $R_0$ and $P(t)$ in a numerically stable way. These approaches presume access to the monodromy matrix, which makes them intractable for large-scale LTP systems. Our approach builds up truncated Floquet bases (columns of $P(t)$), which will serve for the construction of reduced order models.

Floquet transformations are useful yet costly to determine for large-scale LTP systems. The basic transformation follows by noting that the principal fundamental solution matrix, $Φ(t)$, for the homogenous LTP system, $\dot{x}(t) = A(t)x(t)$, has special structure: $Φ(t) = P(t)e^{R_0 t}$ where $P(t) = P(t + T)$ is nonsingular and periodic for all $t$ and $R_0$ is a constant matrix. As one may note in (3), the (time-dependent) basis defined in the columns of $P(t)$ can be used to transform a non-homogeneous LTP system into a corresponding time-invariant system that may be more amenable to analysis (and reduction). However, the computational bottleneck one faces in direct implementation lies in evaluation of the principal fundamental solution matrix over a period, $Φ(t)$, followed by (explicit) computation of the matrix logarithm of the monodromy matrix, $Φ(T)$. Evaluation of the monodromy matrix is computationally intractable even for modest state space dimension if approached via solution of the associated initial value problems (notwithstanding that the idea still is proposed from time to time); computation of the matrix logarithm is numerically delicate but recent advances have made computation of matrix functions less daunting. Nonetheless, difficulties persist for large dimension.

Recent numerically stable approaches to Floquet transformation are built on Fourier spectral approximation for $P(t)$, making the transformation computationally tractable for modest order. The main computational innovation of these approaches lies in reformulating the constraint defining $R_0$ and $P(t)$ as an equivalent boundary value problem,

$$\dot{P}(t) = A(t)P(t) - P(t)R_0,$$

where $R_0$ is constant and $P(0) = P(T) = I$. The columns of $P(t)$ span an $n$-dimensional invariant subspace of the linear map $L = \frac{d}{dt} - A(t)$. Indeed, if $(ω, w)$ is an eigenpair for the matrix $R_0$, then $L(p) = λp$ for $p(t) = P(t)w$. The relationship that this bears to an underlying eigenvalue/invariant subspace problem is at the heart of the extension to a large scale setting. Through an orthogonal change of basis, $R_0$ may be taken to be upper triangular, and so the leading columns of $P(t)$ would then span a family of nested (time-dependent) $T$-periodic subspaces that are invariant for $L$. This remains true even if $P(t)$ is truncated so that only leading columns are retained and this in turn suggests a relationship to the Arnoldi decomposition. The main algorithmic development to be described in this talk provides an analog to the implicitly restarted Arnoldi algorithm or its Krylov-Schur counterpart, enabling us to isolate potentially low dimensional invariant subspaces of particular interest. Invariant subspaces are selectively targeted for convergence through careful selection of shifts, analogous to the implicitly shifted Arnoldi/Krylov-Schur algorithms. We call this approach a truncated Floquet-Arnoldi decomposition in part because of the similarity with the standard Krylov-Arnoldi decomposition in building up recursively, projection bases.
Complex block Jacobi method with application to PGEP

Erna Begović Kovač and Vjeran Hari

Faculty of Chemical Engineering and Technology, University of Zagreb
matematika.fkit.hr/erna

Abstract

We study the global convergence of the block Jacobi method for Hermitian matrices. Let \( A \in \mathbb{C}^{n \times n} \) be Hermitian and let

\[ \pi = (n_1, n_2, \ldots, n_m), \quad n_1 + n_2 + \cdots + n_m = n, \quad n_r \geq 1, \quad 1 \leq r \leq m, \]

be an integer partition of \( n \). Then \( \pi \) defines block-matrix partition of \( A \). Block Jacobi method on \( A \) is an iterative process

\[ A^{(k+1)} = U_k^* A^{(k)} U_k, \quad k \geq 0, \]

where the transformations \( U_k \) are unitary elementary block matrices of the form

\[ U_k = \begin{bmatrix}
I & U_{i_ki_k}^{(k)} & U_{i_kj_k}^{(k)} & & \cdots & & \\
& I & & & & & \\
& & I & & & & \\
& & & I & & & \\
& & & & I & & \\
& & & & & I & \\
& & & & & & I \\
\end{bmatrix} \]

In the \( k \)th step of the method matrix \( U_k \) is chosen to annihilate blocks \( A_{i_ki_k}^{(k)} \) and \( A_{j_kj_k}^{(k)} \), \( i_k \neq j_k \). Block Jacobi method is defined by a pivot ordering, the ordering in which the off-diagonal blocks of \( A \) are annihilated. We are interested in cyclic pivot orderings where during any \( M = m(m-1)/2 \) successive steps of the method every off-diagonal block is annihilated exactly once.

The results that will be presented are a continuation of the work from [1] and [2]. We prove the global convergence of the block Jacobi method for Hermitian matrices for a large class of generalized serial pivot orderings. This class of orderings is introduced in [2] and it is significantly enlarging the well known and commonly used class of weak wavefront orderings. Our generalized serial pivot orderings take pivot positions column-to-column (either from left to right, or from right to left) or row-to-row (either from the top down or from the bottom up), while inside each column/row pivot positions come in an arbitrary ordering.

The convergence results are phrased in the “stronger form”:

\[ \text{off}(A') \leq c_n \text{off}(A), \tag{1} \]

where \( A' \) is the matrix obtained from \( A \) after one full cycle, \( c_n < 1 \) is a constant depending only on \( n \), and \( \text{off}(\cdot) \) stands for the off-norm of a matrix. To prove that inequality (1) holds for any generalized serial pivot ordering we use several techniques with the emphasis on the theory of the complex block Jacobi operators.

Off-diagonal part of a block matrix \( A \) can be represented by a vector \( a \). A Jacobi operator \( J \) is associated with one cycle of a cyclic block Jacobi method. It is obtained as a product of \( M = m(m-1)/2 \) Jacobi annihilators that are ordered according to the specified pivot ordering. Applying \( J \) to a vector \( a \), i.e. the relation \( a' = J a \), can be interpreted as a special block Jacobi method,
where the initial matrix \(A\) and the matrix \(A'\) obtained after one cycle of the Jacobi method are associated with the vectors \(a\) and \(a'\), respectively. It appears that showing
\[
\|J\|_2 \leq c_n < 1
\]
is equivalent to showing (1).

Further on, using the theory of complex Jacobi operators, the result is generalized so it can be used for proving the convergence of more general Jacobi-type processes for other eigenvalue problems. We consider the global convergence problem of a block Jacobi method for solving the positive definite eigenvalue problem (PGEP)
\[
Ax = \lambda Bx, \quad x \neq 0,
\]
where \(A\) and \(B\) are Hermitian matrices and \(B\) is positive definite.

A block Jacobi method for PGEP is an iterative process of the form
\[
A^{(k+1)} = Z_k^* A^{(k)} Z_k, \quad B^{(k+1)} = Z_k^* B^{(k)} Z_k, \quad k \geq 0,
\]
where \(Z_k, k \geq 0\), are unitary elementary block matrices. We prove the global convergence of the PGEP complex block Jacobi method under the class of generalized serial strategies. Precisely, we show that
\[
S(A^{(k)}, B^{(k)}) \to 0, \quad \text{as } k \to \infty,
\]
where
\[
S(A, B) = (\text{off}^2(A) + \text{off}^2(B))^{\frac{1}{2}}.
\]
Finally, all presented results can be extended to the corresponding quasi-cyclic strategies.

References


Greedy Low-rank Connectome Regression

Peter Benner, Sergey Dolgov, Kameron Decker Harris, and Patrick Kürschner

Computational Methods in Systems and Control Theory
Max Planck Institute for Dynamics of Complex Technical Systems
www.mpi-magdeburg.mpg.de/benner

Abstract

Recovering brain connectivity from tract tracing data is an important computational problem in the neurosciences. Mesoscopic connectome reconstruction was previously formulated as a structured matrix regression problem [3], but existing techniques do not scale to the whole-brain setting. The corresponding matrix equation is challenging to solve due to large scale, ill-conditioning, and a general form that lacks a convergent splitting. Conventional low-rank methods for large-scale matrix equations are not able to solve this problem. Therefore, we propose a greedy low-rank algorithm for the connectome reconstruction problem in very high dimensions.

Structural connectivity refers to the synaptic connections formed between axons (outputs) and dendrites (inputs) of neurons, which allow them to communicate chemically and electrically. We represent such networks as a weighted, directed graph encoded by a nonnegative adjacency matrix $W$. The network of whole-brain connections or connectome is currently studied at a number of scales: Microscopic connectivity catalogues individual neuron connections, but currently is restricted to small volumes due to difficult tracing of convoluted geometries [4]. Macroscopic connectivity refers to connections between larger brain regions and is currently known for a number of model organisms [2]. Mesoscopic connectivity [7] lies between these two extremes and captures projection patterns of groups of hundreds to thousands of neurons among the $10^6$–$10^{10}$ neurons in a typical mammalian brain. Our focus is presenting and profiling an improved algorithm for connectome inference. By developing scalable methods as in this work, we hope to enable the reconstruction of high-resolution connectomes in these diverse organisms.

Building on previous work [3, 5], we present a scalable method to infer spatially-resolved mesoscopic connectome from tracing data. The algorithm approximates the solution by a sequence of rank-one updates which exploit the sparse and positive definite problem structure. This algorithm is based on the work by Kressner and Sirkovich [6]. Its adaptation to the connectome regression problem lead to a number of challenges. We design judicious stopping criteria and employ efficient solvers for the three main sub-problems of the algorithm, including an efficient GPU implementation of the Galerkin refinement step that alleviates the main bottleneck for large datasets. The performance of the method is evaluated on three examples: an artificial “toy” dataset and two whole-cortex instances using data from the Allen Mouse Brain Connectivity Atlas [8]. This resource is one of the most comprehensive publicly available datasets.

We find that the method is significantly faster than previous methods and that moderate ranks offer good approximation. This speedup allows for the estimation of increasingly large-scale connectomes across taxa as these data become available from tracing experiments.

The work to be presented is published in [1].
References


Hodge Laplacians and Random Walks on Simplicial Complexes

Michael T. Schaub, Austin R. Benson, Paul Horn, Gabor Lippner, and Ali Jadbabaie

Department of Computer Science, Cornell University
http://www.cs.cornell.edu/~arb/

Abstract

Any time-homogeneous finite state Markov chain can be interpreted as a random walk on a graph: the states of the Markov chain are the nodes of the graph, and transitions occur between connected nodes via an appropriately weighted edge. This close relationship between Markov chains and graphs has led to a broad adoption of diffusion-based algorithms in data science, including methods for clustering, ranking, and dimensionality reduction. Key to the success of many such algorithms is a link between random walks, the structure of data, and the spectral theory of the graph Laplacian, a matrix that encodes the structure of a graph. Specifically, the graph Laplacian $L = D - A$, (1)

where $A$ is the adjacency matrix of the (undirected) graph, and $D$ is the diagonal matrix of node degrees ($D_{ii} = \sum_j A_{ij}$). The matrix $P = AD^{-1}$ is sometimes called the random walk matrix, and can be related to the Laplacian through a normalization or scaling by degrees:

$L := LD^{-1} = I - P$. (2)

Graphs are a special case of more general mathematical objects called (abstract) simplicial complexes. Intuitively, a simplicial complex encodes multi-way relationships between nodes. More formally, we say that a simplex is some subset of nodes, and a simplicial complex $C$ is a set of simplices such that if $x \in C$ and $y \subseteq x$, then $y \in C$. A graph can then be thought of as a simplicial complex where each simplex contains at most two nodes (either nodes or edges). From ideas in algebraic topology, simplicial complexes have a hierarchy of associated Laplacian operators called Hodge Laplacians [1]. For simplicity, we will consider a simplicial complex $C$ where each simplex connects at most three nodes and (keeping the analogy to graphs) call the size-3 simplices triangles, the size-2 simplices edges, and the size-1 simplices nodes. Then the Hodge Laplacian of interest is

$L_1 = B_0^T B_0 + B_1 B_1^T$, (3)

where $B_0$ and $B_1$ are signed edge-to-node and triangle-to-edge incidence matrices. Although the standard graph Laplacian is often written as in Eq. 1, it can be written in a similar form ($L = 0^T 0 + B_0 B_0^T$, with 0 the zero matrix; $L$ is actually a “zeroth-order” Hodge Laplacian, and $L_1$ in Eq. 3 the “first-order” Hodge Laplacian).

Since simplicial complexes can describe richer sets of relationships than a graph, they are increasingly used to model data and analyze complex systems [2]. However, very little is known connecting the Hodge Laplacian to any notion of a Markov chain [3, 4]. In this work, we bridge this gap by showing that an appropriate normalization of the Hodge Laplacian matrix is in fact connected to a random walk on the simplices (with details in [5]). Importantly, these random walks are intimately connected to the topology of the simplicial complex, just as random walks on graphs are related to the topology of the graph. This serves as a foundational step towards incorporating Laplacian-based analytics or diffusions for higher-order interactions and facilitates the translation of the large toolbox of network science for graphs to simplicial complexes.
Our core idea is to use a framework of “matrix liftings” to connect normalized Hodge Laplacians to random walks. In linear algebraic terms, a lifting matrix is simply

\[ V^T = [I \quad -I] \]  

(4)

One can check that applying \( V \) to a scaled version of \( \mathcal{L} \) results in a column-stochastic matrix:

\[ \frac{1}{2}\mathcal{L} = \frac{1}{2} \begin{bmatrix} P & I \\ I & P \end{bmatrix}. \]

(5)

This idea of lifting to a stochastic matrix is what lets us connect the Hodge Laplacian to a stochastic matrix. Specifically, with natural diagonal matrices \( D_1, D_2, \) and \( D_3, \) we first form a normalized version of the Hodge Laplacian:

\[ \mathcal{L}_1 := D_2B_1^TD_1^{-1}B_1 + B_2D_3B_2^TD_2^{-1} \]

(6)

Then this matrix has a similar lifting — for some stochastic matrix \( \hat{P}, \)

\[ \frac{1}{2}\mathcal{L}_1V = \hat{P}, \]

(7)

While a bit mysterious at first, the idea of a lifting naturally arises from dealing with simplex orientation in algebraic topology.

In addition to providing a missing link between Hodge Laplacians and random walks, the normalization and lifting ideas lead to a couple of nice properties. First, the span of the columns of \( V \) is an invariant subspace of the map \( \hat{P}, \) so the spectra are related:

\[ \lambda \left( \frac{1}{2}\mathcal{L}_1 \right) \subset \lambda \left( \hat{P} \right). \]

(8)

Second, the normalized Hodge Laplacian in Eq. 6 leads to a normalized version of the celebrated Hodge decomposition:

\[ \mathbb{R}^{n_1} = \text{im}(B_2) \oplus D_2^{-1}\text{im}(D_2B_1^T) \oplus D_2^{-1}\ker(\mathcal{L}_1), \]

(9)

where \( n_1 \) is the number of edges (size-2 simplices) in the simplicial complex and \( \oplus D_2^{-1} \) denotes the union of orthogonal subspaces with respect to the inner product \( \langle x, y \rangle = x^TD_2^{-1}y. \)

Finally, I will show two data applications to illustrate how our methodology incorporates higher-order topology into data analysis. First, we develop embeddings of edge flows and trajectory data as a higher-order generalization of diffusion maps to create low-dimensional embeddings of the trajectories of ocean drifters. Second, we create variant of personalized PageRank for edges in a simplicial complex to analyze the “role” certain edges play with respect to the global topology.

**References**


Abstract

Diffusion processes and random walks on graphs are important tools for tackling a wide variety of problems arising in Network Science, such as network exploration, consensus algorithms, community detection, clustering, node centrality computations, and many others. Mathematically, these techniques involve an initial value problem (heat equation) associated with the graph Laplacian or the discrete-time Markov chain associated with the normalized Laplacian. Typically, these matrices are sparse (reflecting the sparsity of the underlying network), and one speaks of “local” operators and dynamics.

Recently, the use of fractional Laplacians has been proposed as a tool for solving a range of problems arising in the analysis of networks. In this approach, the usual graph Laplacian $L$ is replaced by a “fractional” power, $L^\alpha$, with $\alpha \in (0,1)$. For an irreducible network, this is usually a full matrix. The corresponding normalized operator generates a random walk with nonzero probability of transitions (“jumps”) from one graph node to any other node, including “far away” ones. These nonlocal diffusion processes have been found to provide a more efficient alternative to the standard, local ones in the exploration of certain networks and in other applications.

Computationally, dealing with such nonlocal operators presents various challenges. Besides the fact that the fractional Laplacian is a full matrix, thus prohibitively expensive to compute for large networks, another difficulty is the fact that the function $f(x) = x^\alpha$ is not analytic at $x = 0$ for $\alpha \in (0,1)$. Since the graph Laplacian is a singular matrix, this indicates that polynomial approximation methods cannot be expected to perform well, therefore rational approximation methods should be used.

In this talk I will discuss some of the computational issues associated with nonlocal diffusion and random walks generated by fractional Laplacians. Furthermore, the extension of the notion of fractional Laplacian to the case of directed networks (digraphs) will be addressed. In addition, I will present some novel applications of the fractional digraph Laplacian.

This is joint work with Daniele Bertaccini (Università di Roma “Tor Vergata”), Fabio Durastante (Istituto per le Applicazioni del Calcolo “Mauro Picone”, Naples), and Igor Simunec (Scuola Normale Superiore).
Efficient preconditioners for MINRES in Convex Constrained Optimization

Luca Bergamaschi, Jacek Gondzio, Ángeles Martínez, John Pearson and Spyridon Pougkakiotis

Department of Civil Environmental and Architectural Engineering, University of Padua (Italy)

www.dmsa.unipd.it/~berga

Abstract

When the Interior Point Method (IPM) is applied to Linear or Quadratic Constrained optimization problems, a sequence of possibly large and sparse Karush-Kuhn-Tucker (KKT) linear systems have to be solved. The matrices involved have a block form like

\[ A = \begin{bmatrix} Q & A^T \\ A & -D \end{bmatrix} \]  

(1)

where \( Q \) is usually a symmetric positive definite \( n \times n \) matrix, \( A \) is the \( m \times n \) rectangular matrix of constraints and we assume \( m < n \). Finally \( D \) is a positive semidefinite diagonal matrix, which takes into account regularization. These matrices have a saddle-point character and they are strongly indefinite. When iterative solvers are elected for the solution of such linear systems, due to the large size of the most challenging problems, they must be accelerated by preconditioners which mimic the block structure of the problem. Among these we quote the Constraint Preconditioner [1] which is very effective in reducing the number of iterations for medium size problems.

When tackling linear optimization problems, matrix \( Q \) is diagonal and hence the linear system \( A \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \) can be reduced to the normal equations (NE in short) linear system

\[ (AQ^{-1}A^T + D) y = AQ^{-1}b_1 - b_2. \]  

(2)

System (2) is symmetric and positive definite hence the Conjugate Gradient method can be successively employed provided that a good preconditioner is available. Unfortunately the entries of diagonal matrix \( Q \), as the IPM iteration proceeds, tend to partition into two subset \( S \) and \( T \) of the indices 1, 2, \ldots, \( n \) such that, at IPM iteration \( k \)

\[ q_{ii}^{(k)} = \begin{cases} O(\mu_k) & \text{if } i \in S \\ O(\mu_k^{-1}) & \text{if } i \in T \end{cases} \]  

where \( \mu_k \rightarrow 0 \) as \( k \rightarrow \infty \) [4]. This causes a sudden deterioration of the condition number of \( AQ^{-1}A^T + D \).

The aim of this talk is to discuss a preconditioner for the normal equation, based on a selective dropping of some rows of \( A \) depending on the magnitude of the elements \( q_{ii} \). In other words, at each IPM step \( k \), we first define as an approximation of \( Q^{-1} \) a diagonal matrix \( E_k \) whose entries are as follows

\[ E_{ii} = \begin{cases} 0 & \text{if } q_{ii}^{-1} < C \min\{\mu_k, 1\} \\ q_{ii}^{-1} & \text{otherwise} \end{cases} \]  

(3)

with \( C \) a suitable constant, and define as the preconditioner for the NE system the (exact) Cholesky factorization of

\[ P_{NE} = AEA^T + D \equiv LL^T. \]  

(4)
The matrix $P_{NE}$ acts as a preconditioner for PCG applied on the normal equations. The dropping strategy just described allows to factorize exactly $P_{NE}$ also in cases where factorization of the original matrix $AQ^{-1}A^T + D$ reveals prohibitive for both storage and computing time.

In order to construct a preconditioner for the augmented system matrix (1), we first note that matrix $Q$ is generally no longer diagonal. We define $\tilde{Q} = \text{diag}(Q)$ and, consequently,

$$E_{ii} = \begin{cases} 0 & \text{if } \tilde{q}_{ii}^{-1} < C \min\{\mu_k, 1\} \\ \tilde{q}_{ii}^{-1} & \text{otherwise} \end{cases}$$

(5)

and $P_{NE}$ as in (4) with this new definition of $E$. To accelerate the MINRES [3] Krylov subspace solver we then employ a symmetric positive definite block diagonal preconditioner of the form:

$$P_{AS} = \begin{bmatrix} \tilde{Q}^{-1} & 0 \\ 0 & P_{NE} \end{bmatrix},$$

(6)

in the solution of the augmented system.

Spectral analysis of the preconditioned matrices in both cases will be provided. Extensive numerical results on large linear and quadratic programming problems will demonstrate the robustness and the efficiency of the proposed approach [2].

References


The Sad Truth about Linear Algebra Software

Henrik Barthels, Christos Psarras, Paolo Bientinesi

Department of Computing Science, Umeå Universitet
hpac.rwth-aachen.de

Abstract

From a survey of many scientific applications that heavily rely on linear algebra computations, we make the observation that there exists a vast disconnect between the developers of linear algebra libraries and the target users.

On one side, the numerical linear algebra and the HPC communities put a great deal of effort in the identification and optimization of functionality that can serve as a set of building blocks for end users. Examples are the various matrix factorizations and linear system solvers, as well as the work that goes into the implementation of kernels such as the matrix-matrix and the sparse matrix-vector products. This set of building blocks is supported by highly sophisticated routines and libraries that take advantage of specific sizes, properties, and structures of the problem, as well as specific features of the target processor, including the instruction set, data types, and different paradigms of parallelism. In this respect, the attention to detail is almost obsessive, with the ultimate goal of making the computation as accurate and/or as fast as possible.

On the other side, many of the matrix problems encountered in actual applications are more involved than the operations supported by libraries, and do not have a direct counterpart in the building blocks. A few examples follow.

• Stochastic quasi-Newton approaches have been proposed for the solution of extremely large least-squares problems [1]. They rely on the computation of a sequence of SPD matrices \( \{B_k\} \) computed as

\[
B_k := \frac{k}{k-1} \left( B_{k-1} - B_{k-1}A^T W_k \left( (k-1)I + W_k^T AB_{k-1} A^T W_k \right)^{-1} W_k^T A B_{k-1} \right),
\]

where \( A \) is the input matrix, and \( W \) is a block Kaczmarz matrix.

• The Kalman filter [2] has countless applications (and variants) in statistics and control theory. In its basic form, the filter performs multiple cycles of the assignments

\[
K_k := P_k^b H^T (HP_k^b H^T + R)^{-1}; \quad x_k := x_k^b + K_k (z_k - H x_k^b); \quad P_k^a := (I - K_k H) P_k^b,
\]

where \( P \) and \( R \) are definite and semi-definite matrices, respectively.

• In order to attenuate artifacts that occur in measurements of brain activity, the Exponential Transient Excision Algorithm (ETEA) is used [3]. In practice, a sparse signal \( x^{(k+1)} \) is iteratively computed as

\[
x^{(k+1)} := A \left[ B^T B + A^T R^T L (Rx^{(k)} R) \right]^{-1} B^T BA^{-1} y,
\]

where matrices \( A \) and \( B \) are square symmetric banded Toeplitz, \( R \) is upper bidiagonal, \( L \) is a smoothing function, and \( y \) is the noisy input data.

In all these cases, users have to make careful decisions on how to decompose the target expressions in terms of the available building blocks, keeping track of how properties and structure propagate, looking for opportunities to avoid redundant computations, creating and exposing as much parallelism as possible, and more. This is a time consuming process that requires expertise in both numerical linear algebra and high-performance computing, and only a minority of end users decide to embark on it. Instead, most turn to one of those programming languages (e.g., Matlab) that offer a high-level interface to linear algebra, thus making it possible to code matrix expressions
such as those in the examples above with extreme ease. This talk aims to answer the question of how efficiently these programming languages compute such expressions.

To this end, we selected seven programming languages (possibly in combination with specific libraries) that allow users to write linear algebra programs at the same level of abstraction as that normally used on a blackboard. The languages are Matlab, Octave, Julia, R, C++ with Eigen, C++ with Armadillo, and Python with Numpy. We then designed a benchmark consisting of simple expressions, each one exposing one particular optimization opportunity that a human expert would take advantage of, and that an “ideal” linear algebra compiler should implement.\footnote{Indeed, several of the benchmark expressions, if performed with scalars in place of matrices, would be optimized by any modern compiler.} Examples include a) mapping onto existing kernels, b) exploitation of matrix structure & properties, c) inference of properties, d) matrix chain algorithm \cite{Barthels2018}, e) partial access and computation, f) loop-invariant code motion, g) common subexpression elimination.

By comparing the execution time with that of handcrafted C reference code, we were able to determine whether or not a language implements a specific optimization. In general, the results paint a fairly grim picture: Most optimizations are ignored by most languages, thus suggesting that despite the high level of sophistication of the building blocks, it is not uncommon for these languages to lose a factor of 10x or even 100x when compared with expert implementations \cite{Fabregat-Traver2013,Fabregat-Traver2013a,Fabregat-Traver2013b}.

This investigation should not be interpreted as a comparison or a competition among languages. Instead, our goal is to expose the core challenges related to the efficient computation of matrix expressions, and to provide the developers of programming languages with clear guidelines to improve their products.

References


On the Characterization of Linear GMRES Convergence and its Stability with respect to Compact Perturbations

Jan Blechta

Faculty of Mathematics, Chemnitz University of Technology
https://www-user.tu-chemnitz.de/~blej/

Abstract

In this work we are concerned with the convergence theory of the GMRES method for operators on a Hilbert space. As a special case, the presented theory applies also to matrices on a finite dimensional vector space. In particular, we are interested in the behavior of the GMRES method for operators which exhibit linear convergence and how the behavior changes under general (large, non-symmetric) compact perturbations; this is quantified in terms of singular values of the perturbation. A preliminary version of the result has been published in the thesis [2, Appendix III.B]; a largely extended version is in preparation for publication [1].

For an operator \( A \) on a Hilbert space, we consider the equation \( Ax = b \) and its GMRES approximations \( x_k \) with residuals \( r_k := b - Ax_k \), i.e., \( \| r_k \| = \min_{p \in P_k, p(0) = 1} \| p(A)r_0 \| \). We consider operators which exhibit linear convergence, i.e., for operator \( A \) there exists \( M_A < 1 \) such that

\[ \| r_k \| \leq M_A \| r_{k-1} \| \]  

holds for all \( k \in \mathbb{N} \) and all initial residuals \( r_0 \).  \( \text{(1)} \)

The well-known sufficient condition for \( M_A < 1 \) that zero is not contained in the numerical range (field of values) of \( A \) is due to Elman’s bound \([4]\)

\[ M_A \leq \sqrt{1 - \frac{\nu_A^2}{\|A\|^2}} \]

and the improved bound by Starke \([8]\), Eiermann, and Ernst \([3]\)

\[ M_A \leq \sqrt{1 - \nu_A \nu_{A^{-1}}} \leq \sqrt{1 - \frac{\nu_A^2}{\|A\|^2}}, \]

where \( \nu_A \) is the distance of the numerical range of \( A \) to the origin. It turns out that \( \nu_A > 0 \) (which is equivalent to \( \nu_{A^{-1}} > 0 \)) is a necessary condition for \( M_A < 1 \). It is an interesting observation that \( \nu_A = 0 \) does not imply the presence of zero in the numerical range of \( A \) in the infinite-dimensional, non-self-adjoint case. In fact, we construct an example of unitary \( A \) which exhibits strictly monotone convergence, i.e., \( \| r_k \| < \| r_{k-1} \| \) or \( \| r_{k-1} \| = 0 \) holds for all \( k \in \mathbb{N} \) and all initial residuals \( r_0 \), but for which \( M_A = 1 \), i.e., there is no \( M_A < 1 \) such that (1) holds.

A problem which has received much attention in the literature \([6, 3, 7]\) is the convergence of \( A = \lambda I + C \) with compact \( C \). We are interested in the more general case \( A = B + C \) with \( M_B < 1 \), i.e., such \( B \) that exhibit linear convergence. We prove that

\[ \frac{\| r_k \|}{\| r_0 \|} \leq \prod_{j=1}^{k} \left( M_B + (1 + M_B) \| A^{-1} \| \sigma_j(C) \right) \quad \text{for all } k \in \mathbb{N} \]
where \( \sigma_j(C) \) are the singular values of \( C \) in decreasing order. If \( C \) is of \( p \)-Schatten class for some \( 1 \leq p < \infty \), i.e., \( \| C \|_p^p := \sum_{i=1}^{\infty} \sigma_j(C)^p < +\infty \), then the last estimate implies

\[
\left( \frac{\| r_k \|}{\| r_0 \|} \right)^{\frac{1}{p}} \leq M_B + k^{-\frac{1}{p}} (1 + M_B) \| A^{-1} \| \| C \|_p \quad \text{for all } k \in \mathbb{N}
\]

This means that the average reduction by the factor \( M_B < 1 \) is slowed down by a superlinearly quickly vanishing delay with the rate \( k^{-1/p} \). This result can be seen as a generalization of Moret’s result [6], where only the case \( B = \lambda I \) was covered. Such a situation implies \( M_B = 0 \) and results in superlinear convergence, as the average reduction is bounded by the term proportional to \( k^{-1/p} \).

We also obtain the asymptotic result

\[
\limsup_{k \to \infty} \frac{\| r_k \|}{\| r_{k-1} \|} \leq M_B.
\]

(2)

It is known that the asymptotic reduction factor, i.e., the left-hand side of (2), depends only on the part of the spectrum \( \sigma(B) \) of non-zero capacity, i.e., \( \sigma(B) \) without its isolated points; see [7]. We can remedy this by removing isolated eigenvalues of finite multiplicity from \( \sigma(B) \) by means of spectral projections for the price of including them in the perturbation \( C \). Thus one is able to handle even \( B \) such that \( M_B = 1 \) if projecting out some finite-dimensional eigenspaces yields \( M_{BS} < 1 \).

Furthermore, we explore a more general class of operators \( B \), the operators which exhibit \( s \)-step linear convergence, i.e., \( B \) with \( s \in \mathbb{N} \) and \( M_{B,s} < 1 \) such that

\[
\| r_k \| \leq M_{B,s} \| r_{k-s} \| \quad \text{holds for all } k \geq s \text{ and all initial residuals } r_0.
\]

As an example consider indefinite self-adjoint \( B \). For such \( B \) one has \( M_B = 1 \) but \( M_{B,2} < 1 \) as long as \( B \) is invertible.

Last but not least, we show how the results apply to the analysis of certain problems in numerical PDEs. A simple example is the convection-diffusion problem, where preconditioning by the Laplacian yields \( B \) coming from the diffusion, which is self-adjoint and positive and thus \( 0 < M_B < 1 \), and a compact \( C \) coming from the convection. A more complex application is the numerical solution of the Navier-Stokes equations. Nearly every solution method is essentially a compact perturbation of a Stokes solver. In the thesis [2, Chapter III] we provided a novel analysis of the pressure convection-diffusion (PCD) preconditioner [5].

References


Low Complexity Computation of the Pseudo-Inverse of a Digraph Laplacian

Daniel Boley

Computer Science and Eng., University of Minnesota
www.cs.umn.edu/~boley

Abstract

Introduction. The pseudo-inverse of the graph Laplacian matrix is often the basis to compute many aggregate quantities related to graphs. Recently it has been shown that a non-symmetric Laplacian derived for a strongly connected directed graph can be used to obtain many similar properties for a directed graph. In [2], it was shown how the pseudo-inverse of the Laplacian with the right scaling leads directly to the average one-way hitting times and average round-trip commute times for a random walk on the digraph. The symmetric part of the pseudo-inverse is a Gram matrix yielding an embedding in Euclidean space where the distance squared between two nodes is directly proportional to the round-trip commute time. It was shown in [7, 3] how this same pseudo-inverse can be used to obtain a third order tensor with entries \(N(i,j,k)\) equal to the average number of visits to intermediate node \(j\) in a random walk starting at node \(i\) and ending at node \(k\), as well as the probability \(P(i,j,k)\) of passing node \(j\) at all in those same random walks. All these quantities lead directly to a ranking of nodes based on their proximity to other nodes (random walk centrality), random-walk betweenness measures, and the centrality with respect to propagation of trust or influence through a social network [4].

In all these cases, the quantities of interest can be computed directly from the the pseudo-inverse of the digraph Laplacian. But this computation itself presents a challenge. A naive computation based on the SVD would cost \(O(n^3)\) which would be prohibitively expensive for even modestly sized graphs. Using mix of a careful ordering of the vertices and carefully crafted preconditioners for iterative methods, a recent result [5, 6] showed it is theoretically possible to find this inverse in almost linear time per column. But these results involve very special methods whose practical implementation remains open.

Here we discuss a complexity bound for the computation of the Moore-Penrose pseudo-inverse that depends entirely on "off-the-shelf" iterative matrix methods well-established in numerical computation. Using these methods, we can arrive at a cost bound that is essentially linear in the number of edges in the graph augmented by a coefficient related to how easily the graph splits. For strongly connected digraphs that satisfy a power-law distribution of the degrees and that are not easily split, this leads to methods to obtain the pseudo-inverse that is almost quadratic in the dimension, i.e., constant time per matrix entry, which is the best one can attain. While the methods are not the most modern ones to accomplish the task and hence might not be the most efficient, they enjoy relatively straightforward bounds on the time complexity. Hence it is a good example of a theoretical cost bound which has a corresponding practical algorithm.

Algorithm. In this note we use a collection of "off-the-shelf" iterative algorithms to construct a procedure to compute the pseudo-inverse with theoretical complexity proportional to the number of edges in the graph. In particular, we form the “Random Walk” graph Laplacian \(L = \text{Diag}(\pi)(I - P)\), where \(P\) is the matrix of transition probabilities for the random walk, and \(\pi\) is the vector of stationary probabilities. We can then obtain the desired pseudo-inverse by the formula [2]

\[
L^+ = \left[\frac{R_1}{n}1_T\right] (L_{1:n-1,1:n-1})^{-1} \left[\frac{-1}{n}1\right],
\]
where $R_1 = (I_{n-1} - 11^T/n)$. The expensive steps are to obtain the vector $\pi$ and then to compute the inverse $(L_{1:n-1,1:n-1})^{-1}$. The former can in principle be obtained by a matrix power method or by solving a system of linear equations (since the eigenvalue is known), in much the same manner as obtaining the Pagerank vector (countless references). The latter can be obtained by a matrix iterative method under suitable conditions. In both cases, we show that these matrices belong to a special class for which the methods, with suitable modifications, can be guaranteed to converge linearly with rates that related to how difficult it is to cut the graph into disconnected pieces. Each iteration has a constant number of matrix-vector products, the step that dominates the total cost. The cost of a matrix-vector product is proportional to the number of nonzeros in the matrix, which is the number of edges in the digraph. Graphs enjoying a power-law degree distribution have a number of edges which is often close to linear in the number of nodes, leading to a quadratic total expected cost to obtain the entire pseudo-inverse. Even if the algorithms used are not the most efficient for the purpose, the convergence theory is straightforward leading to a simple theoretical complexity bound that corresponds directly to computational experience.

Using off-the-shelf matlab code on toy synthetic examples illustrates the complexity behavior. Here the synthetic graph is created using Albert-Barabasi preferential attachment [1].

<table>
<thead>
<tr>
<th>dim.</th>
<th>time</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>$\pi$</td>
<td>$L_{1:n-1,1:n-1}^{-1}$</td>
</tr>
<tr>
<td>2048</td>
<td>21.58</td>
<td>8.24</td>
</tr>
<tr>
<td>4096</td>
<td>42.62</td>
<td>38.14</td>
</tr>
<tr>
<td>8192</td>
<td>82.80</td>
<td>190.31</td>
</tr>
<tr>
<td>16384</td>
<td>185.12</td>
<td>627.72</td>
</tr>
</tbody>
</table>

**References**


Block smoothers in multigrid methods for structured matrices

Matthias Bolten

Fakultät für Mathematik und Naturwissenschaften, Bergische Universität Wuppertal

http://www.math.uni-wuppertal.de/~bolten/

Abstract

Multigrid methods [7] are optimal solvers for important classes of matrices, e.g., matrices stemming from the discretization of elliptic PDEs using finite differences. For a symmetric positive definite matrix $A \in \mathbb{R}^{n \times n}$ the variational theory for multigrid methods, see, e.g., [5], formulates two sufficient conditions for convergence: A smoothing property and an approximation property. The latter requires the existence of a $\beta > 0$ such that for all errors $e \in \mathbb{R}^n$ we have

$$\|Te\|_A^2 \leq \beta \|e\|_A^2,$$

the (pre-)smoothing property asks for an $\alpha > 0$ such that for all errors $e \in \mathbb{R}^n$ we have

$$\|S^\nu e\|_A^2 \leq \|e\|_A^2 - \alpha \|S^\nu e\|_A^2.$$  

Here $S$ is the iteration matrix of the smoother and $T$ is the iteration matrix of the coarse grid correction. This convergence theory has been widely used for structured matrices, as well.

Structured matrices like Toeplitz matrices appear in many applications, e.g., in the discretization of constant coefficient PDEs or in image processing, and can further serve as models for more complex problems, involving complicated geometries or non-constant coefficients. Multigrid methods for these matrices have been studied extensively, e.g., in [6] for Toeplitz matrices and in [1] for matrix algebras, like the circulant algebra.

To speed up multigrid methods block smoothers can be used. These smoothers relax a subset of unknowns rather than a single unknown and result in better smoothing. For some (non-elliptic) problems they are necessary to define working multigrid methods, e.g., the Vanka smoother for Stokes. Here we solely consider blocking to improve the smoothing behavior, at the price of more floating point operations that are necessary. Because of the better smoothing behavior, more aggressive coarsening can be applied. On modern computer architectures the combination can lead to shorter time-to-solution, as parallelism and vectorization can be exploited more easily [2]. To study the behavior of these smoothers, we extended the established local Fourier analysis (LFA) [3]. Nevertheless, LFA only provides convergence estimates for a fixed number of levels.

The theory for multigrid for Toeplitz matrices provides V-cycle convergence results rather than results for a fixed number of levels, only. It already covers more aggressive coarsening [4], yet it lacks an extension for more complex smoothers. In most cases simple smoothers are considered, e.g., Richardson smoothers with optimal relaxation weights in [1]. We analyze block smoothers for Toeplitz matrices by interpreting Toeplitz matrices as block Toeplitz matrices. Using certain properties of the symbol, that is matrix-valued in this interpretation, we are able to derive a simple to check sufficient condition for (1): Given the constant matrix valued symbol of the smoother $\bar{M}$ we demand that

$$\min(\lambda(\bar{M})) \geq \max(\lambda(D(\theta)^{-1} - (I + \alpha D(\theta))^{-\frac{1}{2\nu}} D(\theta)^{-1})),$$

where $D(\theta)$ is a diagonal matrix related to the block symbol of the original Toeplitz matrix interpreted as block Toeplitz matrix.

We will present the steps that are necessary to derive (1) from the sufficient condition and discuss the underlying ideas.
References


A Preconditioner based on Sparsified Nested Dissection and Low-Rank Approximation

Erik G. Boman, Leopold Cambier, Chao Chen, Eric Darve, Siva Rajamanickam, Ray S. Tuminaro

Center for Computing Research, Sandia National Labs, NM 1
www.sandia.gov/~egboman

Abstract

We propose a new algorithm for the fast solution of large, sparse, symmetric positive-definite linear systems, SpaND — Sparsified Nested Dissection [1]. It is based on nested dissection, sparsification and low-rank compression. After eliminating all interiors at a given level of the elimination tree, the algorithm sparsifies all separators corresponding to the interiors. This operation reduces the size of the separators by eliminating some degrees of freedom but without introducing any fill-in. This is done at the expense of a small and controllable approximation error. The result is an approximate factorization that can be used as an efficient preconditioner.

Many sparse direct solvers have recently incorporated low-rank matrix compression schemes to improve their complexity and run-time. The most common approach is to use fast (but approximate) matrix operations on the dense submatrices in a multifrontal or supernodal solver. Examples include MUMPS and PasTix, which use block low-rank operations (BLR). Our approach is different. Instead of first forming large Schur complements for the interfaces and then compress them, we “sparsify” the interfaces during the elimination, effectively reducing the size of the Schur complements. We show empirically that for 3D PDE problems, the top separator size is typically reduced from $O(N^{2/3})$ to $O(N^{1/3})$ using our method. Note that since SpaND has a user parameter for the factorization accuracy, it can be used both as a solver and as a preconditioner. Our focus is on its use as a preconditioner, which typically gives the best time to solution.

Our SpaND method was inspired by the Hierarchical Interpolative Factorization method [2], but we provide several important improvements and extensions:

- Our method is fully algebraic, no underlying mesh is needed. We do partitioning and ordering on the graph of the matrix.
- We introduce a diagonal block scaling that significantly improves stability.
- We use orthogonal transforms instead of interpolative operations, again improving stability and robustness.
- We show SpaND is guaranteed to never break down and the matrix stays symmetric positive-definite (SPD) throughout the process for SPD input matrices.

The computational cost in SpaND is dominated by the rank-revealing factorization (RRQR) used in the “compression” or “sparsification” step. We believe this algorithm is well suited for GPUs and other accelerators. At the bottom levels, there are many small QR factorizations. At the higher levels, there is less parallelism but the dense submatrices become larger.

1Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energys National Nuclear Security Administration under contract DE-NA-0003525.
We perform several numerical experiments to evaluate this algorithm on SPD problems. We demonstrate that a version using orthogonal factorization and block-diagonal scaling takes fewer Krylov (CG) iterations to converge than previous similar algorithms on a wide range of problems. Test problems include SPD problems from the SuiteSparse collection and also ill-conditioned matrices from ice sheet simulations of Antarctica. We evaluate the algorithm on some large problems and show it exhibits near-linear scaling. The factorization time is roughly $O(N)$ and the number of iterations grows slowly with $N$, so the total time to solution is roughly $O(N \log N)$.

The initial algorithm and implementation is limited to the SPD case. However, the approach should extend to the nonsymmetric case with some small modifications. We show several options to extend it to general (nonsymmetric) matrices via approximate LU factorization.

References


Compact Representations of Structured Quasi-Newton Update Matrices

Johannes J. Brust, Sven Leyffer, and Cosmin G. Petra

Mathematics and Computer Science, Argonne National Laboratory

www.johannesbrust.com

Abstract

Quasi-Newton methods, which date to the early 1960’s [1], efficiently approximate the Hessian matrix of $2^{\text{nd}}$ derivatives or the Hessian inverse, of a nonlinear objective function $f(x) : \mathbb{R}^n \to \mathbb{R}$. Classical quasi-Newton methods are formulated as recursive update formulas, which use low-rank (rank-1 or -2) updates to generate the next approximating matrices. Therefore, pioneering work, such as [2], focused on updating matrix factorizations with low-rank modifications. Modern methods are often based on the so-called compact representation, which represents the sequence of low-rank updates as matrix multiplications. In particular, at the previous Householder Symposia (HHXX and HHVII) work on the compact representation of the Broyden class of quasi-Newton updates and block generalized BFGS updates were contributed in [4, 5], respectively. Recently, in [6], two novel structured BFGS (Broyden-Fletcher-Goldfarb-Shanno) quasi-Newton update formulas were introduced for problems in which only one part of the Hessian matrix is approximated and where the other part is assumed to be known. Problems with this structure are found in multivariate optimization. In this presentation we propose the compact representation of the recent recursive structured quasi-Newton updates from [6].

In particular, the structured updates are for problems in which the objective function is formulated as the sum $f(x) = g(x) + u(x)$, where the matrix $\nabla^2 g(x)$ of $2^{\text{nd}}$ derivatives is known, but the matrix $\nabla^2 u(x)$ is unknown. Defining the known derivatives at a current iteration as $K_{k+1} \equiv \nabla^2 g(x_{k+1})$ and the vectors $s_k \equiv x_{k+1} - x_k$, $u_k \equiv (\nabla u(x_{k+1}) - \nabla u(x_k)) + K_{k+1} s_k$ one of two structured quasi-Newton updates (called S-BFGS Plus) is:

$$A_{k+1} = A_k - \frac{1}{s_k^T (K_{k+1} + A_k)s_k} (K_{k+1} + A_k) s_k s_k^T (K_{k+1} + A_k)^T + \frac{1}{s_k^T u_k} u_k u_k^T. \quad (1)$$

The formula in (1) is similar to the widely used BFGS update. Specifically, if $K_{k+1} = 0$, i.e., all $2^{\text{nd}}$ derivatives are unknown, the expression in (1) is the same as the BFGS formula (cf. [3] for properties of BFGS). But since it is desirable to include additional derivative information when it is available, the update in (1) is typically different from BFGS. Because the next matrix $A_{k+1}$ not only depends on the previous $A_k$, but also on $K_{k+1}$, unwinding the recursive relation in (1), to uncover a matrix product representation, is a challenge. Our talk describes the analysis to achieve this goal.

In brief, to develop the compact representation of $A_{k+1}$ we first eliminate the explicit appearance of $K_{k+1}$ in (1) by noting that only the matrix-vector product $v_k \equiv K_{k+1} s_k$ is required in the definition of the recursion. Subsequently, as is common in quasi-Newton compact representations (see for example [4]), we collect the sequence of vectors which appear in the recursion as columns of matrices. Specifically, we define the matrices $S_k \equiv [s_0, \cdots, s_k]$, $V_k \equiv [v_0, \cdots, v_k]$, and $U_k = [u_0, \cdots, u_k]$. The products $S_k^T V_k$ and $S_k^T U_k$ are decomposed as sums of strictly lower-triangular, diagonal, and strictly upper-triangular matrices. Therefore these products are written as $S_k^T V_k = L_k^V + D_k^V + T_k^V$, and $S_k^T U_k = L_k^U + D_k^U + T_k^U$. Using a proof by induction, we obtain the compact matrix representation, which is equivalent to the recursive formula from (1):

$$A_{k+1} = A_0 - [Q_k \ U_k] \left[ \begin{array}{cc} L_k^V + D_k^V + (L_k^V)^T & S_k^T A_0 S_k \ L_k^U - D_k^U \end{array} \right]^{-1} \left[ \begin{array}{c} Q_k^T \ U_k \end{array} \right], \quad (2)$$

130
where $Q_k = V_k + S_k K_k$. At first sight the compact representation appears computationally more expensive, because it requires a solve with a linear system. Nevertheless, in many applications the compact representation is straightforwardly modified to include only solves of small linear systems. In particular, in limited-memory quasi-Newton methods, instead of the entire history of updates, only a few vectors $\{q_i, u_i, v_i\}_{i=(k-m+1)}^k$ are stored (with $m$ as small as 5), resulting in solves with linear-systems of sizes around $10 \times 10$ (see [3]). When $n$ is large the computational cost of solving such small systems becomes negligible in relation to the other computations that depend on $n$. Additionally, the compact representation enables efficient re-initializations of $A_{k+1}$, by allowing initial matrices which depend on $k$, as for instance $A_0 = A_0^{(k)}$. Conventionally, the initial matrix is chosen as a multiple of the identity matrix, i.e., $A_0^{(k)} = \sigma_k I$, where the scalar $\sigma_k > 0$. In a limited-memory strategy, with memory parameter $m \ll n$, the compact representation from (2) with $[q_{k-m+1}, \cdots, q_k, u_{k-m+1}, \cdots, u_k] \equiv \Psi_k \in \mathbb{R}^{n \times 2m}$ and corresponding matrix $M_k \in \mathbb{R}^{2m \times 2m}$ has the form:

$$A_{k+1} = \sigma_k I - \left|\begin{array}{c|c}
\Psi_k \\
\hline
M
\end{array}\right|^{-1} 
\left|\begin{array}{c}
\Psi_k^T
\end{array}\right|$$

(3)

In this case the compact representation enables efficient matrix-vector multiplies, matrix solves, and eigendecompositions without having to form the full matrix $A_{k+1}$. Specifically, these computations are done efficiently by using the factorized form on the right-hand-side of (3). In contrast, the recursive formula from (1) normally requires forming and storing full matrices at each iteration to perform similar computations. Thus for practical large-scale optimization methods the compact representations of quasi-Newton matrices are used in state-of-the-art software.

To summarize, in our presentation we propose the compact representation of two structured BFGS quasi-Newton formulas, introduced in [6]. The proposed representations reduce to the widely used Broyden-Fletcher-Goldfarb-Shanno compact representation when no information about 2nd derivatives is available, but otherwise efficiently incorporates this information.

References


Regularization parameter selection rules for unconstrained and constrained $\ell^p - \ell^q$ minimization

Alessandro Buccini, Mirjeta Pasha, and Lothar Reichel

Department of Mathematics and Computer Science, University of Cagliari
https://sites.google.com/site/alessandobuccini/

Abstract

Inverse problems arise naturally in many fields of science and engineering. We are faced with an inverse problem whenever we need to recover an unknown signal $x \in X$ from some measured data $b \in Y$, with $b = f(x)$, where $X$ and $Y$ are Hilbert spaces, and $f$ is a known or unknown function. We will consider the case in which $X = \mathbb{R}^n$, $Y = \mathbb{R}^m$, and $f$ is a linear function, i.e., we wish to solve the linear system of equations

$$\mathbf{Ax} = \mathbf{b},$$

(1)

where $A \in \mathbb{R}^{m \times n}$, $x \in \mathbb{R}^n$, and $b \in \mathbb{R}^m$. We are interested in the case where $A$ is severely ill-conditioned, i.e., when the singular values of $A$ decay to 0 gradually and with no significant gap. Moreover, we will assume, as often happens in real application, that the right-hand side $b$ is not available, but only a perturbed version of it, denoted by $b^\delta$, is at our disposal. Problems of this kind are often referred to as discrete inverse problems; see, e.g., [6] for a discussion on ill-posed inverse problems.

Let $A^\dagger$ denote the Moore-Penrose pseudo-inverse of $A$. The perturbations in $b^\delta$, often referred to as noise, and the ill-conditioning of $A$ imply that the least squares solution $A^\dagger b^\delta$ is a poor reconstruction of the solution of (1). To compute an accurate approximation of the desired solution $x^\dagger = A^\dagger b$ we need to resort to regularization methods. These methods substitute the original ill-posed problem with a nearby well-posed one whose solution is a good approximation of the first. A regularization method that has attracted considerable interest in the recent years is the $\ell^p - \ell^q$ minimization; see, e.g., [7] and references therein. An accurate approximation of the solution of (1) can be obtained by minimizing either its $\ell^0$-norm or the $\ell^0$-norm of $Lx^\dagger$. However, the minimization of the $\ell^0$-norm is a NP-hard problem. We are then interested in the solution of (2) as a relaxation of the $\ell^0$-norm minimization problem.

We have to determine three parameters in (2): $p$, $q$, and $\mu$. The choice of $p$ is determined by the type of noise. When the noise is Gaussian, i.e., when $b^\delta = b + \eta$ where $\eta$ is a vector whose
entries are realization of a random variable with Gaussian distribution, it is well-known that a good choice is $p = 2$. On the other hand, when the noise that affects the data is impulsive, i.e., it corrupts completely only some entries of $b$ and leaves unchanged the others, the choice $p < 1$ usually provides accurate reconstructions. The choice of $q$ is determined by how sparse we wish $Lx$ to be. From the numerical experiments it can be seen that, if $Lx$ is known to be sparse, the lower the $q$ the more accurate the reconstruction; see [3] for a discussion. The parameter $\mu$ is the most tricky to determine and an imprudent choice may lead to poor reconstructions. In this talk we will discuss how to efficiently solve (2) for any $0 < p, q \leq 2$ (see [7]) and how to determine the regularization parameter $\mu$ both in the case where the noise is Gaussian and its norm is known (see [3]) and when its norm is not known (see [2, 4]). More in details, when the noise is Gaussian and its norm is known we can use the discrepancy principle to compute the parameter $\mu$, i.e., we can look for the parameter $\mu$ such that

$$\|Ax^* - b^\delta\|_2 = \tau \delta,$$

where $\delta > \|b - b^\delta\|_2$ is an estimate of the norm of the noise and $\tau > 1$ is a user-supplied constant. On the other hand when only the type of the noise is known, but no information on its norm its available we will use the Cross Validation (CV) and Generalized Cross Validation (GCV) to determine the regularization parameter. Moreover, we will construct a modification of the CV (MCV) to determine the regularization parameter. Both the CV and GCV determine the parameter that is able to better reconstruct missing data, our proposal will determine the regularization parameter that determines solutions more stable with respect to the loss of data.

In many real problems it is known that the desired solution lies in the non-negative orthant. Thus, we will discuss how to solve the non-negatively constrained version of (1), i.e., how to solve

$$x^* = \arg\min_{x \geq 0} \frac{1}{p} \|Ax - b^\delta\|_p^p + \frac{\mu}{q} \|Lx\|_q^q,$$

(3)

where the inequality is meant element-wise. To this aim we will combine the Modulus-based method constructed in [1] for the solution of the constrained Tikhonov problem with the algorithm described in [7] to compute a solution of (3): see [5]. We will show how, in practice, imposing this constraint substantially improves the quality of the reconstructed solutions.

References


Sparse Matrices Beyond Solvers: Graphs, Biology, and Machine Learning

Aydın Buluç

Computational Research Division, Lawrence Berkeley National Laboratory
Computer Science Division, University of California Berkeley
https://people.eecs.berkeley.edu/~aydin/
https://passion.lbl.gov/

Abstract

Solving systems of linear equations have traditionally driven the research in sparse matrix computation for decades. Direct and iterative solvers, together with finite element computations, still account for the primary use case for sparse matrix data structures and algorithms. These sparse “solvers” often serve as the workhorse of many algorithms in spectral graph theory and traditional machine learning. Examples include spectral clustering, variants of PageRank, support vector machines, and logistic regression.

In this talk, I will be highlighting some of the emerging use cases of sparse matrices outside the domain of solvers. These include graph computations outside the spectral realm, computational biology, and emerging techniques in machine learning. A recurring theme in all these novel use cases is the concept of a semiring on which the sparse matrix computations are carried out. By overloading scalar addition and multiplication operators of a semiring, we can attack a much richer set of computational problems using the same sparse data structures and algorithms. This approach has been formalized by the GraphBLAS effort [2].

Any given non-trivial program in these non-traditional application domains often uses several sparse matrix and vector operations. However, the performance critical portion is often the multiplication of a sparse matrix with a different object, which can be another sparse matrix, a dense matrix, or a sparse vector. In some cases, the matrices can be implicit and are represented as a product or Kronecker product of other matrices [3]. Algorithms for these operations have been understudied due to their historical lack of prevalence in numerical linear algebra.

I will illustrate one example application from each problem domain, together with the most computationally demanding sparse matrix primitive required for its efficient execution. The examples in the graph domain include graph traversals, bipartite graph matching (which ironically has applications in sparse solvers) [11], connected components [12], maximal independent sets, betweenness centrality, triangle counting and enumeration, personalized PageRank, and locally-biased clustering. The examples in the computational biology domain include many-to-many sequence comparisons [13], protein clustering [4], and sequence-to-graph alignment. On the machine learning front, we studied graphical model construction [9], graph kernels [3], and Markov clustering [4].

For each of these three application domains (graphs, biology, and machine learning), there are numerous other problems that can be efficiently tackled by the sparse matrix techniques. I hope that my talk will stimulate more research into those problems. Some of the most exciting areas where sparse matrices on semirings can make a huge impact is on the intersection of these domains, with examples ranging from graph convolutional networks [1] to semi-supervised learning on graphs. I am hoping to briefly touch at least one of these areas in my talk.

Our research efforts helped develop communication-avoiding algorithms for most of the performance-critical sparse-matrix primitives [6, 7]. We have efficiently implemented them on the largest supercomputers on earth and achieved significant performance benefits as predicted by the theory

135
due to reduced communication costs. We have also developed work-efficient shared-memory and GPU-based algorithms in order to scale these problems to Exascale architectures that will often have accelerators and/or many cores on each node [10, 14].

Our approach of using sparse matrix primitives to attack a diverse set of data analytics problems provide a sustainable path of scaling those problems into future architectures thanks to the efforts of two larger communities: one is the community of GraphBLAS enthusiasts who continuously develop new linear-algebraic formulations of data analytics problems [5] and the other is a growing subset of the HPC community who is developing new algorithms and data structures for the sparse matrix primitives [8]. Finally, there is also a growing set of software libraries that implement the aforementioned sparse linear algebraic primitives efficiently on various architectures, including SuiteSparse::GraphBLAS, Combinatorial BLAS, and GraphBLAST.

References


Fast Evaluation and Approximation of the Gauss-Newton Hessian Matrix for the Multilayer Perceptron

Chao Chen, Severin Reiz, Chenhan Yu, Hans-Joachim Bungartz, and George Biros

Oden Institute for Computational Engineering and Sciences, University of Texas at Austin
https://users.oden.utexas.edu/~chao/index.html

Abstract

This talk is concerned with fast entry-wise evaluation of the Gauss-Newton Hessian (GNH) matrix for the multilayer perceptron (MLP). Our algorithm can be used to construct approximations of the GNH matrix for solving linear systems and eigenvalue problems, which are useful for training and analyzing neural networks [1, 2, 3], for selecting training data to minimize the inference variance [4], for network pruning [5], for probabilistic inference [6], and so on.

Consider the MLP with $L$ fully-connected layers and $n$ data pairs $\{(x_i^0, y_i)\}_{i=1}^n$, where $y_i$ is the label of $x_i^0$. Given input data point $x_i^0$, the output of the MLP is computed via the forward pass:

$$x_i^\ell = s(W_{\ell} x_i^{\ell-1}), \quad \ell = 1, \ldots, L$$

(1)

where $s$ is a nonlinear activation function. For ease of presentation, we assume $x_i^\ell \in \mathbb{R}^d$ for $\ell = 0, 1, 2, \ldots, L$ and no bias parameters in (1). Given a loss function $f(x_i^\ell, y_i)$ for the misfit between a network output and the true label, we define the average loss $F(w) = \frac{1}{n} \sum_{i=1}^n f(x_i^L, y_i)$ as a function of the weights $w = [\text{vec}(W_1), \text{vec}(W_2), \ldots, \text{vec}(W_L)]$ of length $N = d^2 L$.

Gauss-Newton Hessian

Let $Q$ be a block diagonal matrix with $Q_i = \frac{1}{n} \partial_{xx}^2 f(x_i^\ell, y_i)$ (Hessian of the loss function $f(x_i^\ell, y_i)$) being the $i$th block. Let $J_i = \partial_w x_i^\ell \in \mathbb{R}^{d \times N}$ be the Jacobian of $x_i^\ell$ with respect to the weights $w$, and $J$ be the vertical concatenation of all $J_i$. The Gauss-Newton Hessian matrix $H \in \mathbb{R}^{N \times N}$ associated with the loss $F$ with respect to the weights $w$ is defined as

$$H = J^T Q J = \sum_{i=1}^n J_i^T Q_i J_i.$$  

(2)

In this talk, we introduce a fast algorithm for computing an arbitrary entry $H_{km}$ in of the GNH matrix for $k, m \in \{1, 2, \ldots, N\}$, where

$$H_{km} = e_k^T H e_m$$

with $e_k$ and $e_m$ being two canonical bases. Without loss of generality, we assume the network employs the mean squared loss ($Q_i$ is the identity), and therefore, the GNH matrix is $H = \frac{1}{n} J J^T$. Observe that $H_{km} = \frac{1}{n} (J e_k)^T (J e_m)$, and only columns in the Jacobian are required to be computed. The idea is to exploit the feed-forward structure of the MLP, where the gradient is back propagated layer by layer, so the intermediate results effectively form a compressed format of the Jacobian with $O(Nn)$ memory (note $J \in \mathbb{R}^{nd \times N}$). As a result, every column can be retrieved in $O(nd)$ time (a column has $O(nd)$ entries).

To further accelerate the computation of $H_{km}$, we introduce a fast Monte Carlo sampling algorithm. Let $v_k(i)$ denote the sub-vector in the Jacobian’s $k$th column corresponding to the $i$th data point, and therefore,

$$H_{km} = \frac{1}{n} \sum_{i=1}^n v_k(i)^T v_m(i).$$  

138
In the sampling, we draw $c$ (independent of $n$) independent samples $t_1, t_1, \ldots, t_c$ from $\{1, 2, \ldots, n\}$ with a carefully designed probability distribution $P_{km}$ (that can be efficiently computed using the previous compressed format of the Jacobian) and compute an estimator

$$\tilde{H}_{km} = \frac{1}{nc} \sum_{j=1}^{c} \frac{v_k(t_j)^T v_m(t_j)}{P_{km}(t_j)}.$$

We prove $\tilde{H}_{km}$ is an unbiased estimator and $|H_{km} - \tilde{H}_{km}| = O(1/\sqrt{c})$ with high probability. As a result, it requires only $O(n + dc)$ work to compute $\tilde{H}_{km}$ as an approximation.

Overall, our algorithm requires $O(Nn)$ storage, and it computes an entry in the GNH matrix with $O(n + d/\epsilon^2)$ work, where $\epsilon$ is a prescribed accuracy. With the fast algorithm, we can compute the hierarchical-matrix ($\mathcal{H}$-matrix) approximation [7, 8] of the GNH matrix. In particular, we take advantage of the existing GOFMM method [9, 10] to construct the $\mathcal{H}$-matrix approximation through evaluating $O(N)$ entries in the GNH matrix. Once constructed, the $\mathcal{H}$-matrix approximation leads to fast linear algebra operations [11, 12, 13, 14]. For example, the approximation can be factorized in $O(N)$ time for solving linear systems and eigenvalue problems. We show numerical experiments on the cost and the accuracy of the overall framework.

References


Analyzing the Effects of Local Roundoff Error on Predict-and-Recompute Conjugate Gradient Variants

Tyler Chen, Erin Carson, Anne Greenbaum

Applied Mathematics, University of Washington
https://chen.pw

Abstract

In this talk, we discuss the roundoff error analysis of several predict-and-recompute conjugate gradient variants introduced in [Che19]. These variants, like other communication hiding variants, reduce the runtime per iteration by overlapping global communication with computation and other communication. However, in numerical experiments, the predict-and-recompute variants are observed to have better numerical properties than previously studied communication hiding variants such as pipelined CG from [GV14]. This was speculated to be due to the use of a predict-and-recompute scheme, whereby recursively updated quantities are first used as a predictor for their true values and then recomputed directly at a later point in the iteration. Our rounding error analysis provides evidence suggesting that this is indeed the reason for the improved convergence.

The conjugate gradient algorithm (CG) is perhaps the most widely used method for solving a very large linear system of equations $Ax = b$, when $A \in \mathbb{R}^{n \times n}$ is symmetric positive definite. Like other Krylov subspace methods, the CG algorithm iteratively computes approximations $x_k$ of the solution $x = A^{-1}b$ over successive Krylov subspaces. In order to do this, the algorithm maintains and recursively updates several vectors including the “updated residual” $r_k$, which in exact arithmetic is equal to the true residual $b - Ax_k$.

However, in finite precision, these recursively updated vectors will typically deviate from their corresponding true values. In fact, it is well known that the conjugate gradient algorithm is particularly sensitive to rounding errors, and any modification to the CG algorithm will change the numerical behaviour. Specifically, both the rate of convergence (number of iterations to reach a given level of accuracy) and the maximal attainable accuracy of any CG implementation may be severely impacted by carrying out computations in finite precision.

Despite this sensitivity to rounding errors, many mathematically equivalent variants of the CG algorithm have been introduced to run more efficiently on parallel computers; see for instance [CG89, GV14]. Broadly speaking, these variants aim to rearrange the standard CG algorithm in such a way that communication occurs less frequently, or is overlapped with other computations. As a result, the time per iteration of these so called communication hiding methods is reduced on parallel machines. However, the practical use of some of these variants is limited because on many problems the algorithms fail to reach an acceptable level of accuracy, or require so many more iterations to do so that the overall runtime is not decreased.

Maximal Accuracy The predict-and-recompute variants from [Che19] are observed to reach a higher maximal accuracy on problems where other communication hiding variants, such as pipeline CG, fail to reach an acceptable level of accuracy. In order to study this phenomenon, we follow the approach of [Gre97, SvdVF94] and compute an expression for the residual gap $r_k - (b - Ax_k)$ in terms of local rounding errors. For many CG variants, it is observed that the updated residual $r_k$ decreases to much lower than the machine precision, and so the size of the residual gap gives an estimate for the size of the true residual once the algorithm has converged.
In this talk, we discuss how the expression for the residual gap in the predict-and-recompute variants does not amplify certain error terms in the way that other communication hiding variants do. This type of analysis was done by Gutknecht and Strakoš in [GS00] for a three-term CG variant meant to reduce communication costs, and it was shown why the maximum attainable accuracy of that variant was reduced. Similarly, the residual gap of pipelined conjugate gradient variants is studied in [CFYA+18, CRS+18]. However, for some variants, the updated residual $r_k$ may not decrease to well below machine precision, so some care must be taken when interpreting these results.

**Rate of Convergence** In addition to the theory about the maximal accuracy of the CG algorithm in finite precision, there is also (highly nontrivial) theory about the rate of convergence of the CG algorithm in finite precision, due primarily to Greenbaum in [Gre89]. In this analysis, it was shown that for any iteration $k$, a CG implementation run in finite precision will behave like exact CG applied to a larger matrix whose eigenvalues lie in small intervals about the eigenvalues of $A$, provided that (i) the updated residuals approximately satisfy the three term Lanczos recurrence, and (ii) successive residuals are approximately orthogonal. This analogy provides a means to apply results about exact precision CG to finite precision implementations.

In order to study the rate of convergence of the predict-and-recompute variants, we follow the approach of [GLC19] where several of the previously mentioned communication hiding CG variants were analyzed. Specifically, we compute expressions involving local rounding errors for quantities (such as the error in the three term Lanczos recurrence) which play an important role in the analysis of [Gre89]. The form of the expressions we derive for the predict-and-recompute variants provides some intuition for why they seem to exhibit better rates of convergence than other communication hiding variants.

**Outlook** The behaviour of any CG variant in finite precision is still not completely understood. Even so, we hope that our analysis can help inform the development of future high performance conjugate gradient variants.

**References**


Sampled Limited Memory Methods for Massive Least Squares Problems

Julianne Chung, Matthias Chung, J. Tanner Slagel, and Luis Tenorio

Department of Mathematics, Virginia Tech
https://intranet.math.vt.edu/people/mcchung/

Abstract

In this work, we investigate sampled limited memory row-action methods to approximate the solution to the classic least squares (LS) problem. Given $A \in \mathbb{R}^{m \times n}$ with full column rank and $b \in \mathbb{R}^m$, the goal is to approximate the unique solution,

$$x_{LS} = \arg\min_x \|Ax - b\|^2 = (A^\top A)^{-1} A^\top b.$$  \hfill (1)

The LS problem is ubiquitous and core to computational mathematics and statistics. However, for massive problems where both the number of unknown parameters $n$ and the size of the observational dataset $m$ are massive or dynamically growing, standard techniques based on matrix factorization or iterative methods based on full matrix-vector multiplications (e.g., Krylov subspace methods or randomized methods) are not feasible. Problems of such nature appear more frequently and are plentiful, for instance in classification problems, data mining, 3D molecular structure determination, super resolution imaging, and 3D tomography.

Row-action methods such as the Kaczmarz method have emerged as an attractive alternative due to their scalability, simplicity, and quick initial convergence [1]. Basically, row-action methods are iterative methods where each iteration only requires a sample of rows of $A$ and $b$, thus circumventing issues with memory or data access. The most widely-known row-action methods are the Kaczmarz and block Kaczmarz methods, where only one row or one block of rows of $A$ and $b$ are required at each iteration. Various extensions have been proposed to include random sampling and damping parameters, and many authors have studied the convergence properties of these methods.

To mathematically describe the sampling process, let $W$ be a uniformly distributed random variable on a finite sample space \( \{ W^{(i)} \}_{i=1}^M \) where $\mathbb{E}W W^\top = \beta I$ for some $\beta > 0$. Further, let $W_k \in \mathbb{R}^{m \times \ell}$ denote an i.i.d. sample of $W$. Then, at the $k$-th iteration we denote the available block of rows as $A_k = W_k^\top A$ and $b_k = W_k^\top b$.

We propose row-action method called sampled limited memory for LS (slimLS) to approximate (1), where given an arbitrary initial guess $x_0 \in \mathbb{R}^n$, the $k$-th slimLS iterate is defined as

$$x_k = x_{k-1} - B_k A_k^\top (A_k x_{k-1} - b_k),$$

with

$$B_k = \left( \alpha_k^{-1} C_k + M_k^\top M_k \right)^{-1} \quad \text{and} \quad M_k = [A_{k-r}^\top, \ldots, A_k^\top]^\top.$$

Here $\{C_k\}$ is a sequence of positive definite matrices, $\{\alpha_k\}$ is sequence of damping parameters, and the parameter $r \in \mathbb{N}_0$ is a “memory parameter” where we define $A_{k-r}$ with negative index as an empty matrix. Utilizing a particular sampling scheme, slimLS is a generalization of the damped block Kaczmarz method.

The slimLS method can also be interpreted as a stochastic approximation method [2]. Using the properties of $W$ described above, one can show that LS problem (1) is equivalent to the stochastic optimization problem,

$$\arg\min_x \mathbb{E} \left\| W^\top (Ax - b) \right\|^2.$$  \hfill (3)
Stochastic approximation methods for (3) have the form \( x_k = x_{k-1} - B_k \nabla f_{W_k}(x_{k-1}) \), where \( f_{W_k}(x) = \| W_k^\top (Ax - B) \|^2 \) and \( \{B_k\} \) is some sequence of positive semi-definite matrices. For the particular choice of \( B_k \) defined in (2), we see that the slimLS method is a stochastic approximation method.

Our contributions are as follows.

1. In this work we show that our proposed slimLS method exhibits favorable initial and asymptotic convergence properties. Furthermore, we prove linear convergence of the expected squared error norm up to a “convergence horizon” (compare [3, 4]).

2. We reveal the power of the slimLS method in various numerical examples, where we show that the slimLS method can be used for massive and streaming problems. For instance, in a 3D tomographic reconstruction problem, we illustrate that for matrices of sizes \( A \in \mathbb{R}^{261,121,000 \times 133,432,831} \), slimLS is able to compute satisfying reconstructions on a standard desktop computer. We compared slimLS to methods available on this setting, i.e., stochastic gradient methods, Kaczmarz, and online limited memory BFGS methods. We observe superior convergence of slimLS.

3. We investigate extensions to our method, were \( A \) is ill-conditioned and regularization is required to obtain meaningful reconstructions. We investigate Tikhonov regularization and introduce sampled regularization selection methods, i.e., sampled unbiased predictive risk estimator, sampled generalized cross validation, and sampled discrepancy principle. A major advantage of these methods is that they do not require full knowledge of \( A \) to iteratively estimated the regularization parameter, see [5]. We illustrate our methods on an ill-posed superresolution problem, were we aim to reconstruct a high dimensional image out of a sequence of low resolution images.

4. We further discuss separable nonlinear problems, where the objective function is nonlinear in one (typically small) set of parameters and linear in another (larger) set of parameters, [6]. We describe an approach to integrate the limited-memory sampled Tikhonov method within a nonlinear optimization framework. We illustrate our methods again on an ill-posed superresolution assuming unknown nonlinear motion parameters.

References


On a Spectral Method for “Bipartizing” a Network and Detecting a Large Anti-Community

Anna Concas, Silvia Noschese, Lothar Reichel and Giuseppe Rodriguez

Dipartimento di Matematica e Informatica, Università degli Studi di Cagliari (Italy)
http://bugs.unica.it/~anna/

Abstract

Network analysis aims to identify important nodes in a network and to uncover structural properties of a network such as bipartivity. A bipartite network models the interactions between two different types of objects and involves objects that can be split into two disjoint groups with connections occurring only across, but not within, the two groups. Specifically, a network $G$ is said to be bipartite if the set of vertices $V$, that make up the graph, can be partitioned into two disjoint nonempty subsets $V_1$ and $V_2$ (with $V = V_1 \cup V_2$), such that any edge starting at a vertex in $V_1$ points to a vertex in $V_2$, and vice versa.

Bipartivity is an important topological property that has been studied also as the 2-coloring problem [1]. Indeed, bipartite networks are also known as two-colourable graphs, as only two colours are needed, for example red and blue, to colour their nodes so that red nodes are connected only to blue nodes, and vice versa. Hence, determining if a graph can be colored with 2 colors is equivalent to determining whether or not the graph is bipartite. However, in the majority of real-world situations this “perfect” separation into two colours (or classes more in general), leading to exact bipartite networks, is not always possible and consequently many networks in real applications are only almost bipartite. Hence, there have been some efforts to quantify how much bipartivity a non-bipartite graph has [5, 4] and it is therefore interesting to determine a bipartite approximation of a non-bipartite network, or measure the distance of a non-bipartite graph from being bipartite.

In order to find an approximate solution to the the difficult problem of determining the “closest” bipartite network to a given graph, in the work [2] coauthored by the applicant, it is described how a starting network can be approximated by a bipartite one by solving a sequence of fairly simple optimization problems. In particular, it will be discussed a numerical method developed for determining a “good” bipartization $(V_1, V_2)$, i.e., a bipartization for which the sum of the weights $w_k$ associated with the edges $e_k = (i, j)$ that point from a vertex $v_i$ in $V_1$ to a vertex $v_j$ in $V_2$, or vice versa, is fairly small. The proposed spectral “bipartization” algorithm exploits the spectral structure of a bipartite graph for detecting an approximate bipartization and it determines a node permutation which highlights the possible bipartite nature of the initial adjacency matrix identifying the two disjoint sets of nodes expected. The algorithm is exact whenever it is applied to the adjacency matrix of a bipartite graph, however it has to be considered approximate, or “heuristic”, in the sense that it does not necessarily produce the best possible bipartization. Some computed examples will be explored to illustrate the performance of the presented spectral method.

The same computational procedure may also be used to detect the presence of a large anti-community in a network, i.e., a group of nodes loosely connected internally but with many external links, and identify it. The identification of communities and anti-communities, whose detection is a relevant problem with application in various fields, is predominant in the investigation of the so-called meso-scale structures which represent middle-scale properties in networks.

The performance of the spectral bipartization algorithm when applied to the detection of anti-communities, it will be illustrated by analyzing real networks, both weighted and unweighted.
Considering that bipartivity can be generalized to \( k \)-partivity, some results, related to a work in progress [3], will be given on the generalization of the spectral bipartization method for discovering approximate multipartite structures.

References


Cross Approximation, Column Subset Selection, and Maximum Volume Submatrices

Alice Cortinovis\(^1\), Daniel Kressner, and Stefano Massei

École Polytechnique Fédérale de Lausanne, Switzerland.

Abstract

Given a matrix \( A \in \mathbb{R}^{n \times n} \), an integer \( 1 \leq k \leq n \), and two index sets \( I, J \subset \{1, \ldots, n\} \) of cardinality \( k \), the rank-\( k \) matrix \( A(:,J)A(I,J)^{-1}A(I,:) \) is called a cross approximation, where we assume the \( k \times k \) submatrix \( A(I,J) \) to be invertible. Cross approximations are used, for example, for accelerating computations within the boundary element method, in uncertainty quantification, machine learning (Nyström method), spectral clustering, and low-rank tensor approximation.

In this talk, we will present new developments \([2, 3]\) for cross approximation: new a priori error bounds for the most commonly used method, a greedy algorithm \([1]\), and a new algorithm with guaranteed Frobenius norm error bounds based on an improvement of the column subset selection algorithm by Deshpande and Rademacher \([5]\).

A priori error bounds

If the matrix \( A(I,J) \) is of maximum volume, that is, the absolute value of its determinant is maximum among all \( k \times k \) submatrices of \( A \), then the corresponding cross approximation \( A_{\text{vol}} := A(:,J)A(I,J)^{-1}A(I,:) \) is a quasi-optimal low-rank approximation of \( A \). In particular, the seminal result by Goreinov and Tyrtyshnikov \([7]\) states that

\[
\|A - A_{\text{vol}}\|_{\text{max}} \leq (k + 1)\sigma_{k+1}(A),
\]

where \( \|\cdot\|_{\text{max}} \) denotes the maximum absolute value of the entries of a matrix and \( \sigma_{k+1}(A) \) – the \((k + 1)\)th largest singular value of \( A \) – is the error of the best rank-\( k \) approximation of \( A \) in the spectral norm. Unfortunately, finding the maximum volume submatrix is NP hard, limiting the practical usefulness of this result.

Greedy algorithms for volume maximization are the most common way to obtain a cross approximation. We focus on the adaptive cross approximation \([1]\) algorithm with full pivoting, equivalent to (partial) LU factorization with full pivoting. We show that the obtained rank-\( k \) cross approximation \( A_{\text{greedy}} \) satisfies \([2]\)

\[
\|A - A_{\text{greedy}}\|_{\text{max}} \leq 4^k \cdot \rho_k \cdot \sigma_{k+1}(A),
\]

where \( \rho_k \leq 2\sqrt{k + 1}(k + 1)^{\ln(k + 1)/4} \) is the growth factor of Gaussian elimination with full pivoting \([8]\). If the matrix \( A \) has additional structure, the bound \((1)\) can be improved. For instance, for symmetric positive semidefinite matrices there is no pivot growth and result \((1)\) holds without the \( \rho_k \) factor, recovering a result in \([6]\). For matrices which are diagonally dominant by rows and by columns the exponential factor disappears and we obtain \( \|A - A_{\text{greedy}}\|_{\text{max}} \leq 2(k + 1)^2 \cdot \sigma_{k+1}(A) \).

Improvements of column subset selection algorithm by Deshpande and Rademacher

The column subset selection problem aims at finding \( k \) columns of \( A \in \mathbb{R}^{m \times n} \) that approximate the range of \( A \). It was proved in \([4]\) that sampling column indices \( X = (X_1, \ldots, X_k) \in \)

\(^1\)The work of Alice Cortinovis has been supported by the SNSF research project Fast algorithms from low-rank updates, grant number: 200020_178806.
\{1, \ldots, n\}^k \text{ according to the square volume of the corresponding } m \times k \text{ submatrix } A(:, X) \text{ provides }
\mathbb{E} \left[ \| A - A(:, X) A(:, X)^+ A \|_F^2 \right] \leq (k + 1) \left( \sigma_{k+1}(A)^2 + \ldots + \sigma_m(A)^2 \right), \text{ where } ^+ \text{ denotes the Moore-Penrose pseudo-inverse and the volume of } A(:, X) \text{ is the product of its } k \text{ largest singular values.}

In [5] it was proposed to subsequently select indices \((i_1, \ldots, i_k) =: I\) that minimize, for \(t = 1, \ldots, k\), the conditional expectation \(\mathbb{E}[\| A - A(:, X) A(:, X)^+ A \|_F^2 | X_1 = i_1, \ldots, X_t = i_t]\). It is possible to write the latter quantity in terms of coefficients of the characteristic polynomial of a suitable matrix [5]. This gives a deterministic algorithm to obtain \(I\) that satisfies
\[
\| A - A(I,:)(I,:)^+ A \|_F \leq \sqrt{k+1} \sqrt{\sigma_{k+1}(A)^2 + \ldots + \sigma_m(A)^2},
\]
only a factor \(\sqrt{k+1}\) away from the error of the best rank-\(k\) approximation in the Frobenius norm.

We propose to compute the conditional expectations from the singular values of the involved matrices in order to improve the stability of the algorithm. We attain efficiency by exploiting rank-1 updates of the SVD, and further speeds up (up to 40 times) by slightly relaxing the minimization problem, stopping the search for each \(i_t\) as soon as a “sufficiently good” column is found [3].

We also show that the result (2) leads to improved bounds for CUR approximations of matrices, that is, approximations of the form \(A \approx CUR\), where \(C\) and \(R\) denote \(k\) columns and rows of \(A\), respectively, and \(U \in \mathbb{R}^{k \times k}\). Column subset selection can be applied to \(A\) and \(A^T\) to get the column and row indices, respectively. This can be generalized to tensors in Tucker format.

**New cross approximation algorithm with Frobenius norm error bounds** It was recently proved [9] that sampling submatrices \(A(I,J)\) of \(A \in \mathbb{R}^{n \times n}\) for index sets \(I, J \subset \{1, \ldots, n\}\) of cardinality \(k\) according to the square volume of \(A(I,J)\), yields the existence of a cross approximation \(A_{\text{Frob}}\) that satisfies
\[
\| A - A_{\text{Frob}} \|_F \leq (k + 1) \sqrt{\sigma_{k+1}(A)^2 + \ldots + \sigma_n(A)^2},
\]
By a technique analogous to the one described above for columns subset selection, we subsequently choose pairs \((i_1, j_1), \ldots, (i_k, j_k)\) such that suitable conditional expectations, which can be expressed in terms of coefficients of some characteristic polynomials, are small enough. We obtain an \(O(kn^4)\) algorithm that constructs index sets \((I, J)\) that satisfy (3). To the best of our knowledge, this is the first polynomial-time algorithm for cross approximation that guarantees a bound of the form (3).

**References**


Uniform Invariant Subspace Perturbation Theory

Anil Damle and Yuekai Sun

Computer Science, Cornell University

Abstract

Given a symmetric matrix $A$, a classical problem is to understand how subspaces spanned by collections of eigenvectors of $A$ change as $A$ is perturbed. More specifically, given $A$ and some “small” perturbation $E$ one seeks to bound the differences between invariant subspaces of $A$ and $A + E$. Numerous classical results such as the Davis-Kahan Theorem [4] address this problem and a more thorough overview can be found in [6]. Notably, virtually all these results measure distances between subspaces via metrics related on principle angles.

In this work we develop new “row-wise” bounds on the difference between invariant subspaces. While care must be taken to define row-wise differences in a sensible manner for subspaces, we first illustrate the distinction between row-wise bounds and traditional theory via a simple example. Let $v$ be an eigenvector of $A$ with a simple eigenvalue, similarly let $\hat{v}$ be the corresponding eigenvector of $A + E$ (assuming $E$ is sufficiently small). Classical results focus on bounding

$$\sqrt{1 - (v^T \hat{v})^2},$$

whereas we develop results controlling

$$\min_{s = \pm 1} \|v - s\hat{v}\|_\infty.$$

There are many settings where changes in eigenvectors are well distributed over the entries and the infinity norm measure may be much smaller than the angle between vectors. Many modern spectral methods interpret eigenvectors entry-wise and their performance and analysis are dependent on controlling and understanding entry-wise changes in eigenvectors.

Here, we consider a symmetric square matrix $A \in \mathbb{R}^{n \times n}$ with eigen-decomposition

$$A = V_1 \Lambda_1 V_1^T + V_2 \Lambda_2 V_2^T,$$

(1)

where $\Lambda_1 \in \mathbb{R}^{k \times k}$ is diagonal and contains the $k$ algebraically largest eigenvalues of $A$, $V_1 \in \mathbb{R}^{n \times k}$ represents the associated eigenvectors, $\Lambda_2 \in \mathbb{R}^{(n-k) \times (n-k)}$ contains the remainder of the eigenvalues, and $V_2 \in \mathbb{R}^{n \times (n-k)}$ represents the associated eigenvectors. We order the eigenvalues algebraically (i.e., $\lambda_1 \geq \cdots \geq \lambda_n$) and assume $\lambda_k > \lambda_{k+1}$ so this split is well defined. Given symmetric $E \in \mathbb{R}^{n \times n}$ we denote the eigen-decomposition of $\hat{A} = A + E$ as

$$\hat{A} \equiv A + E = \hat{V}_1 \hat{\Lambda}_1 \hat{V}_1^T + \hat{V}_2 \hat{\Lambda}_2 \hat{V}_2^T,$$

(2)

where the respective matrices have analogous interpretation to (1). The restriction to considering the algebraically largest eigenvalues is not essential, any isolated set is fine, but affords notational simplicity. Additionally, we will always make assumptions on the size of $E$ to ensure associating $V_1$ and $\hat{V}_1$ makes sense. Lastly, $E$ need not be symmetric if we consider the Schur form of $\hat{A}$.

Concretely, given (1) and (2) we are interested in bounding

$$\min_{Q \in O^k} \|V_1 - \hat{V}_1 Q\|_{2,\infty},$$

(3)
where $\mathcal{O}^k$ is the set of $k \times k$ orthogonal matrices and $\| \cdot \|_{2,\infty}$ is the induced matrix norm. Notably, for any matrix $B$ we have that $\|B\|_{2,\infty} = \max_j \|B_{j,:}\|_2$, i.e., the maximum of the 2-norm of the rows. This entry-wise definition of $\| \cdot \|_{2,\infty}$ is why we refer to our perturbation results as row-wise.

We may now state an informal version of our main result; formal statements and detailed proofs may be found in the preprint [3]. Following the above notation, for symmetric $A$, symmetric $E$, and constant $\gamma$ related to the separation of $\Lambda_1$ and $V_2A_2V_2^T$, if $\|E\|_2 \leq \gamma/5$ we have that

$$
\min_{Q \in \mathcal{O}^k} \|V_1 - \hat{V}_1Q\|_{2,\infty} \leq 8\|V_1\|_{2,\infty} \left( \frac{\|E_{2,1}\|_2}{\lambda_k - \lambda_{k+1}} \right)^2 + 2\|V_2E_{2,1}\|_{2,\infty} \frac{\gamma}{\lambda_k - \lambda_{k+1}} + 4\|V_2V_2^TE\|_{2,\infty} \|E_{2,1}\|_2$$

(4)

where $E_{i,j} = V_i^T EV_j$. Importantly, in situations where $E$, $EV_1$, and $V_1$ have relatively uniform row norms (4) can be a significantly tighter bound than that afforded by simply using Davis-Kahan to naively bound (3) using the fact that $\| \cdot \|_{2,\infty} \leq \| \cdot \|_2$.

While (4) is a purely deterministic result, in the course of its proof we uncover a natural starting point for more nuanced probabilistic analysis when $E$ is random. Nevertheless, in [3] we construct a deterministic example to show that in the worst case all three terms in (4) are necessary and any of them may be the dominating factor in the bound. Similarly, we also provide extensive experiments in [3] to illustrate the effectiveness of our bound in predicting behavior when $E$ is random. Our experiments clearly demonstrate our theoretical prediction that in many situations there is no difference between $A$ being rank-$k$ and $A$ having rank greater than $k$—behavior not reflected by prior work considering row-wise metrics.

The most closely related results to our own are those of [1] and [2], though some earlier work explored the single eigenvector case [5]. In certain simple regimes our bounds are in agreement with prior work, but in others we improve on existing results — importantly, as noted above, when $A$ is not rank-$k$ we construct stronger upper bounds. Additionally, much of the aforementioned work is concerned with analyzing random models and often intertwines probabilistic assumptions with the construction of bounds. Our assumptions only involve the eigen-decomposition of $A$ and ensuring $E$ is small enough to associate invariant subspaces of $A$ and $\hat{A}$.

While (4) is illustrative of our results, additional bounds, including for non-normal $A$ and $E$, may be found in [3]. Our results provide new insight into the behavior of invariant subspaces when the associated matrix is perturbed. Showing when our bounds are tight enables us to better understand our bounds and clearly demonstrate where improvements may be made for analysis of specific problems. Additionally, we touch on how certain independence assumptions on $E$ in the random setting are sufficient for constructing tighter bounds. We believe our bounds aid in the analysis of existing spectral algorithms and the understanding of observed practical performance. Specific applications of interest are discussed as are algorithmic paradigms motivated by this work.

References


Multiparameter eigenvalue problems

Bart De Moor

ESAT-STADIUS, dept. EE, KU Leuven, Belgium
bart.demoor@kuleuven.be, www.bartdemoor.be

Abstract submitted to the Householder meeting, June 2020

The eigenvalue problem (EVP) is fundamental and ubiquitous in science and engineering. Its theory, applications and algorithms have been described abundantly. For the multiparame

ter eigenvalue problem (MEVP), our theoretical and algorithmic understanding is much less elaborate (early references include [1] [2]). This deficiency is deplorable as applications of the MEVP are also ubiquitous in science and engineering.

In this contribution, we will discuss theory, algorithms and applications for MEVPs, using a combination of insights from (multi-)dimensional system theory (realization algorithms, see e.g. [3]), algebraic geometry (multivariate polynomials, ideals and varieties, see e.g. [4]), operator theory (shift-invariant model spaces, see e.g. [5] [6]) and numerical linear [7] or polynomial [8] algebra.

The MEVP is to find the non-trivial eigenvectors \( x \in \mathbb{C}^n \) and the eigentuples \( \lambda_i \in \mathbb{C}, i = 1, \ldots, p \) in

\[
(A_0 + A_1 \lambda_1 + \ldots + A_p \lambda_p)x = 0, \tag{1}
\]

where the matrices \( A_i \in \mathbb{R}^{m \times n} \) with \( m \geq n \) contain the problem data. Special cases are the square (Jordan Canonical Form), generalized square (Weierstrass Canonical Form) and rectangular (Kronecker Canonical Form) EVP.

For the sake of clarity of exposition, we consider 5 prototypical cases, each of which starts with one or more (multivariate) polynomial seed equation(s), from which new (equivalent) equations are generated by multiplying them with all monomials of increasing degree. With this forward shift recursion (FSR), we create structured matrices (‘quasi-Toeplitz’), the null spaces of which have special, shift-invariant properties that can be exploited to calculate the solutions of the seed equations via one or several EVPs.

Case 1: Single shift scalar banded Toeplitz matrix: The seed equation is a univariate polynomial \( p(\lambda) \) of degree \( n \) in a single variable \( \lambda \). The FSR generates a banded Toeplitz matrix, the null space of which has the structure of a (confluent) Vandermonde matrix, or (in system theory terms) an observability matrix of a linear time-invariant (LTI) system with a single output. By exploiting the shift-invariant structure, the roots of the characteristic equation (the eigenvalues) can be calculated via realization theory.

Case 2: Single shift scalar Sylvester matrix: The seed problem here is to find the common roots of two univariate polynomials \( p(\lambda) = 0 \) and \( q(\lambda) = 0 \). The FSR generates a Sylvester matrix. Its nullity reveals the number of common zeros and its null space can be shown to be shift-invariant. The common roots are then obtained via a realization algorithm.

Case 3: Single shift block banded Toeplitz matrix: The seed problem is the polynomial matrix eigenvalue problem \( (A_0 + A_1 \lambda + A_2 \lambda^2 + \ldots + A_p \lambda^p)x = 0 \), where the real matrices \( A_i \in \mathbb{R}^{m \times n}, m \geq n \) are given. The FSR for this case generates a block banded Toeplitz matrix. Again, its nullity reveals the number of roots, that can be calculated from several EVPs, by exploiting the block shift structure of the null space via realization theory.

Case 4: Multi-shift scalar Macaulay matrix: The seed equations form a set of multivariate polynomials. We want to find their common roots. The FSR generates a quasi-Toeplitz matrix, called a Macaulay matrix. The nullity equals the Bezout number (i.e. the number of affine zeros
and zeros at infinity), which can be found by exploiting the multi-shift invariant structure of the null space and applying multi-dimensional realization algorithms to it.

**Case 5: Multi-shift block Macaulay matrix:** The seed problem is the polynomial multi-parameter eigenvalue problem. As an example, for $p = 2$, with 2 parameters $\lambda_1$ and $\lambda_2$, it is of the form $(A_{00} + A_{10}\lambda_1 + A_{01}\lambda_2 + A_{20}\lambda_1^2 + A_{11}\lambda_1\lambda_2 + A_{02}\lambda_2^2 + \ldots)x = 0$ with $A_{ij} \in \mathbb{R}^{m \times n}$, $m \geq n$, with an obvious generalization for $p$ variables. The FSR now generates a block Macaulay matrix, the nullity of which corresponds to the number of solutions. We demonstrate how its null space can be modelled as the observability matrix of a $p$-dimensional discrete shift invariant state space model with multiple outputs.

For each of these cases, we describe how, starting from the ‘seed equation(s)’, the FSR generates structured, sparse, quasi-(block)-Toeplitz matrices, the null spaces of which are scalar or vector, single- or multi-shift invariant projective subspaces. They can be ‘modelled’ as observability matrices of (possibly) singular, autonomous, commutative, (multi-)dimensional discrete shift-invariant dynamical systems [9]. Obtaining the null space is an exercise in linear algebra (e.g. via the SVD), while exploiting the (multi-)shift invariant structure leads to several EVPs, that together deliver all the (common) roots of the seed equation(s).

As a special application we discuss the computation of the global minimum of a multivariate polynomial optimization problem, which corresponds to calculating the minimizing root of a MEVP of the form (1). Important engineering examples include the identification of LTI dynamic models from observed data, where a sum-of-squares of the so-called prediction errors is minimized [10]. There is a rich variety of model classes like ARMA(X), Box-Jenkins, etc., which have been described abundantly in the statistical and engineering literature, the identification of which requires the solution of a nonlinear least squares optimization problem. All known algorithms are heuristic (local minima, convergence behavior, etc.). But a crucial observation (see e.g. [11]) is the fact that all these models and the objective function are multivariate polynomial. As a consequence, one only needs to find the minimizing solution of an MEVP of the form (1), a fact that we consider to be a fundamental breakthrough.

### References


Krylov Recycling for Iterative Hybrid Projection Methods with Applications to Streaming Data

Julianne M. Chung, Eric de Sturler, and Jiahua Jiang
Department of Mathematics, Virginia Tech
https:\intranet.math.vt.edu\people\sturler

Abstract

Iterative hybrid projection methods have proven very effective for large linear inverse problems, due to their inherent regularizing properties and the advantage of selecting regularization parameters adaptively. In this talk, we present enhanced Golub-Kahan-based hybrid projection methods that exploit Krylov recycling techniques to efficiently solve a broad class of large inverse problems with streaming data, multiple data sets, or where memory constraints prevent using standard approaches. The latter includes problems with many unknowns and requiring many iterations. Standard iterative hybrid regularization requires all basis vectors for the solution search space to be available. Enhanced hybrid projection methods selectively keep relevant subspaces of the search space (compression). This leads to significant speed-up with solution accuracy comparable to that of standard methods. For streaming data problems or inverse problems with multiple datasets, the described methods efficiently integrate previously computed information in a hybrid framework for faster and better reconstruction. As a further benefit, the proposed methods can incorporate various subspace selection techniques, standard techniques for automatic regularization parameter selection, and they can be applied in an iterative fashion. We present some surprising (desirable) theoretical results, and numerical examples from image processing show the potential benefits of using recycling to enhance hybrid projection methods.

We consider large, linear inverse problems (or sequences of these) of the form,

$$b = Ax_{\text{true}} + \epsilon,$$

where $A \in \mathbb{R}^{M \times N}$ models the forward process, $b$ contains observed data, $x_{\text{true}}$ represents the desired parameters, and $\epsilon$ is noise or measurement error. Given $b$ and $A$, the goal is to compute an approximation to $x_{\text{true}}$. In this talk, we focus on Tikhonov regularization,

$$\min_x \|Ax - b\|_2^2 + \lambda^2 \|x\|_2^2,$$

where $\lambda$ is a (yet-to-be-determined) regularization parameter that balances the data-fit term and the regularization term. Other hybrid regularization approaches of the projected problem are possible too.

Enhanced hybrid projection methods use recycling to extend a given subspace, chosen a priori or selected in previous computations, and compute regularized solutions with automatic regularization parameter selection for the projected problem on the extended subspace. The general approach consists of three steps, which can be used in an iterative fashion and with multiple compressions.

1. Choose $W_{k-1} \in \mathbb{R}^{N \times (k-1)}$ with orthonormal columns to represent the recycle space, derived from a related problem or from solutions and search spaces in previous reconstructions.

2. Given an approximate solution, $x_0$, we use the recycling Golub-Kahan bidiagonalization (GKB) process to extend the solution space to $R([W_{k-1} x_0 \tilde{V}_\ell])$, where $\tilde{V}_\ell$ represents a particular Krylov subspace, typically orthogonal to $R([W_{k-1} x_0])$.

3. Solve the regularized projected problem,

$$\min_{x \in R([W_{k-1} x_0 \tilde{V}_\ell])} \|Ax - b\|_2^2 + \lambda^2 \|x\|_2^2.$$
We proceed as follows. From \([W_{k-1} x_0]\) compute \(W_k\) with \(W_k^T W_k = I\), and compute the thin QR decomposition \(Y_k R_k = AW_k\). Next, let \(\tilde{r} = b - A x_0 - Y_k \zeta\) with \(\zeta = Y_k^T (b - A x_0)\), and start the recycling GKB with \(\tilde{u}_1 = \tilde{r}/\beta_1\), where \(\beta_1 = \|\tilde{r}\|_2\) (note \(\tilde{u}_1 \perp Y_k\)), and iterate:

\[
\begin{align*}
\tilde{\alpha}_j \tilde{v}_j &= A^T \tilde{u}_j - \tilde{\beta}_j \tilde{v}_{j-1} \\
\tilde{\beta}_{j+1} \tilde{u}_{j+1} &= (I - Y_k Y_k^T) A \tilde{v}_j - \tilde{\alpha}_j \tilde{u}_j \\
&= (I - Y_k Y_k^T) A \tilde{v}_j - \tilde{\alpha}_j \tilde{u}_j
\end{align*}
\]  

(4)

(5)

Note that, in exact arithmetic, \(W_k \perp \tilde{V}_j\) without explicit orthogonalization. After \(\ell\) iterations, we have the recycling GKB relation

\[
A \begin{bmatrix} W_k & \tilde{V}_\ell \end{bmatrix} = \begin{bmatrix} Y_k & \tilde{U}_{\ell+1} \end{bmatrix} \begin{bmatrix} R_k & Y_k^T A \tilde{V}_\ell \\ O & B_\ell \end{bmatrix},
\]

(6)

where \([W_k \tilde{V}_\ell]\) and \([Y_k \tilde{U}_{\ell+1}]\) both have orthonormal columns. We can compute regularized solutions of the form \(x = W_k c + \tilde{V}_\ell d\) from the small (projected) Tikhonov problem,

\[
\begin{bmatrix} c \\ d \end{bmatrix} = \arg \min_{c,d} \| \begin{bmatrix} R_k & Y_k^T A \tilde{V}_\ell \\ O & B_\ell \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix} - \begin{bmatrix} \zeta + R_k W_k^T x_0 \\ \beta_1 e_1 \end{bmatrix} \|_2^2 + \lambda^2 \| \begin{bmatrix} c \\ d \end{bmatrix} \|_2^2.
\]

(7)

If memory constraints are an issue, we can repeatedly compress the search space and extend again. For regularized solutions, our approach satisfies the desirable property that if the discarded space corresponds to the small singular values of the projected matrix, the solution computed with compression is a small perturbation of the solution computed without compression.

For streaming problems or problems with multiple data sets, we can block-wise compute recycle spaces, including approximate solutions and important spectral information, that allow us to update approximate solutions very fast and compute accurate solutions for the full problem, even though solving the full problem at once is not feasible (either memory-wise or time-wise).

Related ideas for recycling in iterative methods have been considered in the literature, and we will provide comparisons with alternative approaches. In particular, several alternative approaches enrich the search space, but do not force the search space to remain orthogonal to this enrichment space (or enforce orthogonality of the image spaces). Hence, while the search space is extended, the Krylov space generated is not changed, in contrast with our approach. An important example is the hybrid enriched bidiagonalization (HEB) method that stably and efficiently augments a “well-chosen enrichment subspace” with the standard Krylov basis associated with LSQR [Hansen et al., NLAA 2019]. We will show that for several tomography problems our approach provides much better reconstructions.

In this presentation, we will show some key theoretical properties of our recycling GKB approach, and we will present several numerical examples that show the excellent behavior of our approach in time and in accuracy. We will present results for dynamic or streaming data inverse problems where reconstructions must be updated as data is being collected, and starting from scratch with all data is not feasible. This may arise in applications such as microCT, where immediate feedback may be used to inform the data acquisition process during the scan. Another example occurs in solving reconstruction problems where the projection angles are slightly modified. This might arise in an optimal experimental design framework where the goal is to determine the optimal angles for image formation or in a sampling framework.

This presentation also involves a collaboration with Kirk Soodhalter and Misha Kilmer on a unifying framework for recycling-based iterative methods.
Deep neural networks have been remarkably successful at the difficult tasks of machine translation, object recognition and speech recognition. At its core, a deep neural network may be looked upon as a repeated composition of the following: a linear transformation followed by a special nonlinear map. The linear transformation is typically learned using a stochastic optimization scheme known as stochastic gradient descent. This scheme requires the so-called back propagation phase, where gradients of the differentiable objective are propagated through the network while training the deep neural network. However, vanishing and exploding gradients are two significant obstacles in training deep neural networks, especially in capturing long range dependencies in recurrent neural networks (RNNs). In this talk, I will present a parametrization of the transition matrix of an RNN that allows us to modulate the gradients that arise in its training. Specifically, the transition matrix is parameterized by its singular value decomposition (SVD), which allows us to explicitly track and control its singular values. Efficiency is obtained by using Householder reflectors for representing the left and right singular vector matrices. By explicitly controlling the singular values, our proposed svdRNN method allows us to solve the exploding gradient problem, and mitigate the vanishing gradient problem to a large extent. The SVD parameterization can be used for any rectangular weight matrix, hence it can be easily extended to any deep neural network, such as a multi-layer perceptron. Theoretically, we can demonstrate that our parameterization does not lose any expressive power. Experimental results demonstrate that the proposed framework converges faster, and has better generalization, especially when the depth is large.
Versal Deformations in Numerical Linear Algebra

Andrii Dmytryshyn

Örebro University
https://www.oru.se/english/employee/andrii_dmytryshyn

Abstract
Canonical forms of matrices and matrix pencils, e.g., Jordan and Kronekher canonical forms, are well known and studied with various purposes but reductions to these forms are unstable operations: both the corresponding canonical forms and the reduction transformations depend discontinuously on the entries of an original matrix or matrix pencil. This issue complicates the use of canonical forms for numerical purposes. Therefore V.I. Arnold introduced a normal form, with the minimal number of independent parameters, to which an arbitrary family of matrices $\tilde{A}$ close to a given matrix $A$ can be reduced by similarity transformations smoothly depending on the entries of $\tilde{A}$. He called such a normal form a miniversal deformation of $A$ [1, 2, 3]. Now the notion of miniversal deformations has been extended to general [16, 19] and structured [7, 13, 14] matrix pencils, matrices of bilinear [13] and sesquilinear [14] forms, as well as to matrices under similarity over various fields [4, 18, 19].

In this presentation we will discuss the use of miniveasal deformation for the problems in numerical linear algebra, namely: in investigation of possible changes in canonical forms (eigenstructures), reduction of the unstructured perturbations to structured perturbations, codimension computations, and nearness problems.

Changes in canonical forms (eigenstructures). Miniversal deformations can be used for analyzing changes in the canonical forms (eigenvalues, their multiplicities, minimal indices) of matrices and matrix pencils under perturbations, e.g., for showing if the eigenvalues may coalesce or split apart, appear or disappear. More generally, miniversal deformations can help us to construct closure hierarchy graphs of orbits and bundles [6, 10, 11, 12, 15, 17]. These graphs show which canonical forms the matrices (or matrix pencils) may have in an arbitrarily small neighbourhood of a given matrix (or matrix pencil).

Reduction to structured perturbations. (Mini)versal deformations have or may be forced to have a certain structure, e.g., blocking, sparcity. Therefore the theory of versal deformations provides a possibility to take into account all the possible perturbations of a given matrix or matrix pencil while working only with their versal deformations, i.e. particularly structured matrices or matrix pencils. One of such examples is reduction of a perturbation of matrix polynomial linearization to a linearization of a perturbed polynomial, or in the other words, finding which perturbations of the matrix coefficients of a matrix polynomial correspond to a given perturbation of the entire linearization pencil. For this example we present an algorithm, inspired by the theory of versal deformations, that performs such a reduction [8]. Moreover we also derive the transformation matrices that via strict equivalence transform perturbation of linearizations to the linearization of perturbed polynomials [8].

Codimension computations. The number of independent parameters in a miniversal deformation is equal to the codimension of the corresponding orbit (e.g., similarity orbit for Jordan canonical form, or strict equivalence orbit for Kronekher canonical form) [5, 7, 9]. Therefore computing a miversal deformation automatically provides us with the corresponding codimension.
Nearness problems. Versal deformations may help us to solve various nearness problems for matrix pencils and matrix polynomials, e.g., finding a singular matrix pencil or polynomials nearby a given matrix pencil or polynomial.

References


Multi-linear algebra for approximating high dimensional functions in Bayesian inverse problems

Sergey Dolgov, Karim Anaya-Izquierdo, Colin Fox, Robert Scheichl

Department of Mathematical Sciences, University of Bath, UK
https://people.bath.ac.uk/sd901

Abstract

Many large-scale computer models used in science and engineering are often driven by many parameters that can not be directly observed. Thus, we need to solve an inverse problem to infer model parameters from some incomplete and indirect data to make model predictions. In addition, we need to the quantify the uncertainty of the parameter estimations and the model predictions to assess the confidence of our model. Typical examples can be found in subsurface flows, composite material manufacturing, meteorology, econometric, and beyond.

Inverse problems and uncertainty quantification can be formalised in the Bayesian framework [1]. Considering unknown parameters to be inferred as random variables \( \mathbf{x} = (x_1, \ldots, x_d) \in \mathbb{R}^d \), we introduce some prior probability density \( \pi(\mathbf{x}) \) and the likelihood function of the measured data \( L(y|\mathbf{x}) \). Applying the Bayes law, we use the posterior density \( \pi(\mathbf{x}) \propto L(y|\mathbf{x})\pi(\mathbf{x}) \) to describe the solution to the inverse problem and the associated uncertainty. This way, parameter estimations and uncertainty quantifications can be computed as summary statistics, which are interpreted as solution to the inverse problem and the associated uncertainty. This way, parameter estimations and uncertainty quantifications can be computed as summary statistics, which are interpreted as solution to the inverse problem and the associated uncertainty.

I plan to present our recent research [8] on using tensor decompositions to tackle both challenges altogether. To approximate a high-dimensional function (e.g. \( \pi(\mathbf{x}) \)), we introduce \( n \) points in each variable and collect the nodal values of \( \pi(\mathbf{x}) \) into a \( n \times n \times \cdots \times n \) tensor with elements \( p(i_1, \ldots, i_d) \). The original function can be approximated by e.g. a piecewise linear or polynomial interpolation. However, such a tensor with \( n^d \) elements is impossible to even store directly when \( d \) exceeds tens.

To reduce the complexity, we can arrange the tensor values into a matrix \( \mathbf{P} \in \mathbb{R}^{n \times n^{d-1}} \) and apply the truncated singular value decomposition (SVD) to obtain a rank-1 approximation \( \mathbf{P} \approx \mathbf{U} \Sigma \mathbf{V}^\top \). The key idea now is to regroup the elements of \( \Sigma \mathbf{V}^\top \) into a \( r_1n \times n^{d-2} \) matrix, and apply again a matrix factorisation. The latter factor in the new factorisation can be reshaped similarly, and we can continue reducing the dimension recursively to arrive at the so-called tensor–train (TT) [2] decomposition:

\[
p(i_1, \ldots, i_d) \approx \tilde{p}(i_1, \ldots, i_d) = \sum_{s_1, \ldots, s_{d-1}=1}^{r_1 \cdots r_{d-1}} p^{(1)}(i_1, s_1)p^{(2)}(s_1, i_2, s_2) \cdots p^{(d)}(s_{d-1}, i_d). \tag{1}
\]

Of course, actually using SVD is intractable in high dimensions, since it requires access to all \( n^d \) entries in the first step. However, the TT approximation can be computed efficiently using a much smaller number of evaluations by the so-called TT-Cross algorithm [2]. In contrast to SVD, this is based on the rank-\( r \) skeleton decomposition \( \mathbf{P}(:, \mathcal{J}) \mathbf{P}(\mathcal{I}, \mathcal{J})^{-1} \mathbf{P}(\mathcal{I}, :) \) of a matrix \( \mathbf{P} \in \mathbb{R}^{n \times n} \), where \( \mathbf{P}(:, \mathcal{J}) \) are \( r \) columns of the target matrix \( \mathbf{P} \), specified by some set of indices \( \mathcal{J} \), and \( \mathbf{P}(\mathcal{I}, :) \) are \( r \) rows corresponding to (another) set \( \mathcal{I} \). Clever pivoting strategies, such as the maximal volume principle [3] or the discrete empirical interpolation method [4], provide theoretical and (heuristic)
practical recipes on how to choose quasi-optimal sampling sets $I, J$. If $r$ is much smaller than $n$, drawing only $r$ rows and columns makes this construction much more efficient than SVD. The TT-Cross extends this idea recurrently to higher dimensions. It reconstructs a TT approximation using only $O(dnr^2)$ adaptively chosen samples. Significant enhancements of the TT-cross, e.g. the solution of parametric PDEs [5] and parallel implementation [6], were developed recently.

If both $\pi(x)$ and $G(x)$ can be approximated using low-rank TT decompositions, the expectation $\mathbb{E}_\pi G$ can be computed using a univariate quadrature applied to each of the factors in (1) separately. However, this assumption is often violated in statistics, especially for $G(x)$. For example, to compute the probability that $x_2 \geq x_1$, $G(x)$ must be an indicator function, equal to 1 when $x_2 \geq x_1$ and 0 otherwise. The matrix of its nodal values is upper-triangular and obviously full-rank. This issue escalates dramatically for higher $d$.

We proposed [8] to combine advantages of tensor decompositions and Monte Carlo methods to alleviate this issue. The idea is to compute first a low-rank TT approximation to $\pi(x)$ only, or even to a simplified version thereof, such as $\pi(x)^\beta$ for $\beta < 1$. This enables efficient computation of the Rosenblatt transformation of variables [7]. This procedure computes a map $x = T_\pi(y)$ from reference variables $y \in [0, 1]^d$ to the original space $x \in \mathbb{R}^d$, with $x$ distributed according to the density function $\pi(x)$. In the first step, we compute $x_1$ as the solution to $C(x_1) = y_1$, where $C(x_1)$ is the marginal cumulative distribution function. Continuing recursively, we compute the next component $x_i$ as the solution to $C(x_i|x_1, \ldots, x_{i-1}) = y_i$, where $C(x_i|x_1, \ldots, x_{i-1})$ is the cumulative conditional density function. Computation of $C(x_i|x_1, \ldots, x_{i-1})$ requires the integration of $\pi$ over subsets of variables, but when applied to the TT approximation (1), this procedure breaks down to cheap univariate integration of each TT factor with the same $O(dnr^2)$ total cost per sample.

This TT-Rosenblatt transformation is a powerful technique that can be used in different ways:

- by drawing $y \in [0, 1]^d$ randomly or from a quasi Monte Carlo lattice. This yields realisations of $x$ that can be used as nearly-best proposal samples in Metropolis and Importance Sampling methods. The resulting nearly optimal quadrature points allow to approximate $\mathbb{E}_\pi G$ significantly more accurate than gold-standard MCMC methods, also when $G(x)$ is irregular.

- As an adaptive mesh generator in high dimensions, by introducing a uniform mesh in $y$ and transforming it through $T_\pi$. The latter can aid further approximations of more difficult functions.

I believe that this potentially impactful development, underpinned by simple and efficient linear algebra, will be of interest to the audience of the Householder Symposium.

References


Backward stability of rational eigenvalue problems solved via linearization

Froilán M. Dopico, María C. Quintana, and Paul Van Dooren

Departamento de Matemáticas, Universidad Carlos III de Madrid (Spain)
https://gauss.uc3m.es/fdopico/fdopico.html

Abstract

Rational eigenvalue problems (REPs) are defined as follows: given a regular rational matrix \( R(\lambda) \), that is, a matrix whose entries are rational functions of a complex variable \( \lambda \), the goal is to compute numbers \( \lambda_0 \) and nonzero vectors \( x \) such that \( R(\lambda_0)x = 0 \). The numerical solution of REPs is receiving considerable attention in the last years as a consequence of two facts: (1) they arise directly in a number of applications [8], and (2) they are fundamental in the solution of other nonlinear eigenvalue problems [5], since a first standard step for solving such problems is to approximate them by REPs [6, 7]. The standard, and probably most reliable, method for solving REPs is via linearization [1, 2, 3, 8], i.e., by constructing a matrix pencil that contains all or part of the spectral information of the rational matrix \( R(\lambda) \) and solving the corresponding generalized eigenvalue problem for this pencil with the QZ algorithm, for dense problems, or with Krylov methods adapted to the particular structure of the linearization in the large scale setting [4, 6].

A key difficulty of working with REPs is that they may appear represented in different forms depending on the underlying applications, or on the different methods that may be used for approximating a nonlinear eigenvalue problem by a rational one. Despite this variety of REPs, nowadays there are many ways of linearizing them [2, 3] and of solving efficiently the corresponding generalized eigenvalue problem [4, 6, 7]. In contrast, there are not yet results about the structural backward stability of these algorithms for solving REPs, where “structural” means that the backward errors are attached to the matrix inputs of the problem, that is, to the constant matrices that define the representation of the starting rational matrix.

In this talk, we will present a structured backward error analysis of REPs solved via a wide family of linearizations [2], when the rational matrix is expressed as the sum of a polynomial matrix and a strictly proper rational matrix in state-space form. This representation is a general one, is the standard one in control problems, and captures many interesting problems arising in applications. The presented analysis shows that the poles of the rational matrix may produce a lack of structural backward stability, which can be solved if the rational problem is properly scaled in advance.

References


New computational tools for Koopman spectral analysis of nonlinear dynamical systems

Zlatko Drmač, Igor Mezić and Ryan Mohr

Department of Mathematics, Faculty of Science, University of Zagreb
https://web.math.pmf.unizg.hr/~drmac/

Abstract

Dynamic Mode Decomposition (DMD) is a data-driven spectral analysis technique for a time series. For a sequence of snapshot vectors $f_1, f_2, \ldots, f_{m+1}$ in $\mathbb{C}^n$, assumed driven by a linear operator $A$, $f_{i+1} = Af_i$, the goal is to represent the snapshots in terms of the computed eigenvectors and eigenvalues of $A$. We can think of $A$ as a discretization of the underlying physics that drives the measured $f_i$’s. In a pure data-driven setting we have no access to $A$. Instead, the $f_i$’s are the results of measurements, e.g. computed from pixel values from a high-speed camera recorded video.

No other information on the action of $A$ is available. (In another scenario of data acquisition, $A$ represents PDE/ODE solver (software toolbox) that generates solution in high resolution, with given initial condition $f_1$.)

Such a representation of the data sequence provides an insight into the evolution of the underlying dynamics, in particular on dynamically relevant spatial structures (eigenvectors) and amplitudes and frequencies of their evolution (encoded in the corresponding eigenvalues) — it can be considered a finite dimensional realization of the Koopman spectral analysis, corresponding to the Koopman operator associated with the nonlinear dynamics under study [1]. This important theoretical connection with the Koopman operator and the ergodic theory, and the availability of numerical algorithm [5] make the DMD a tool of trade in computational study of complex phenomena in fluid dynamics, see e.g. [6]. Its exceptional performance motivated developments of several modifications that make the DMD an attractive method for analysis and model reduction of nonlinear systems in data-driven settings. A peculiarity of the data-driven setting is that direct access to the operator is not available, thus an approximate representation of $A$ is achieved using solely the snapshot vectors $f_i$ whose number $m+1$ is usually much smaller than the dimension $n$ of the domain of $A$.

In this talk, we present recent development of numerically robust computational tools for dynamic mode decomposition and data-driven Koopman spectral analysis.

First, we consider computations of the Ritz pairs $(z_j, \lambda_j)$ of $A$, using both the SVD based Schmid’s DMD [5] and the natural formulation via the Krylov decomposition with the Frobenius companion matrix. We show how to use the eigenvectors of the companion matrix explicitly — these are the columns of the inverse of the notoriously ill-conditioned Vandermonde matrix. The key step to curb ill-conditioning is the discrete Fourier transform of the snapshots; in the new representation, the Vandermonde matrix is transformed into a generalized Cauchy matrix, which then allows accurate computation by specially tailored algorithms of numerical linear algebra. Numerical experiments show robustness in extremely ill-conditioned cases. More details can be found in [2], [3].

Secondly, the goal is to identify the most important coherent structures in the dynamic process under study, i.e. after computing the Ritz pairs $(z_j, \lambda_j)$ of $A$ the task is to determine $\ell < m$, the indices $\varsigma_1 < \cdots < \varsigma_\ell$ and the coefficients $\alpha_j$ to achieve high fidelity of the snapshot representations

$$f_i \approx \hat{f}_i \equiv \sum_{j=1}^{\ell} z_{\varsigma_j} \alpha_j \lambda_{\varsigma_j}^{i-1}, \quad i = 1, \ldots, m.$$ (1)
After selecting an appropriate subset \( \{ z_j \} \) of the modes, the key numerical step is solution of a structured linear least squares (LS) problem \( \sum_{i=1}^{m} \| f_i - \hat{f}_i \|_2^2 \rightarrow \min \) for the coefficients \( \alpha_j \) in (1). The coefficients of the representation are determined from a solution of a structured linear least squares problems with the matrix that involves the Khatri-Rao product of a triangular and a Vandermonde matrix. Such a structure allows for a very efficient normal equation based least squares solution, which is used in state of the art computational fluid dynamics (CFD) tools, such as the sparsity promoting DMD (DMDSP). A new numerical analysis of normal equations approach provides insights about its applicability and its limitations. Relevant condition numbers that determine numerical robustness are identified and discussed. Further, a corrected semi-normal solution and the QR factorization based algorithms are proposed. It will be shown how to use the Vandermonde-Khatri-Rao structure to efficiently compute the QR factorization of the least squares coefficient matrix, thus providing a new computational tool for the ill-conditioned cases where the normal equations may fail to compute a sufficiently accurate solution. Altogether, we present a firm numerical linear algebra framework for a class of structured least squares problems arising in a variety of applications besides the DMD, such as e.g. multistatic antenna array processing. All details can be found in [4].

The main message of the talk is that the state of the art numerical linear algebra provides sharp and robust numerical tools for computational analysis of nonlinear dynamical systems.

References


Embedding properties of network realizations of dissipative reduced order models

*Vladimir Druskin, Murthy Guddati, Rob Remis and Jörn Zimmerling*

Mathematical Sciences Department, Worcester Polytechnic Institute
https://www.wpi.edu/people/vdruskin

**Abstract**

Realizations of reduced order models of passive SISO or MIMO LTE problems can be transformed to tridiagonal and block-tridiagonal forms, respectively, via different modifications of the Lanczos algorithm. Generally, such realizations can be interpreted as ladder resistor-capacitor-inductor (RCL) networks. They gave rise to network syntheses in the first half of the 20th century that was at the base of modern electronics design and consecutively to MOR that tremendously impacted many areas of engineering (electrical, mechanical, aerospace, etc.) by enabling efficient compression of the underlying dynamical systems. In his seminal 1950s works Krein realized that in addition to their compressing properties, network realizations can be used to embed the data back into the state space of the underlying continuum problems.

In more recent works of the first author (with Liliana Borcea, Fernando Guevara Vasquez, David Ingerman, Leonid Knizhnerman, Alexander Mamonov, Shari Moscow and Michael Zaslavsky) Krein’s ideas gave rise to so-called finite-difference Gaussian quadrature rules (FDGQR), allowing to approximately map the ROM state-space representation to its full order continuum counterpart on a judiciously chosen grid. Thus, the state variables can be accessed directly from the transfer function without solving the full problem and even explicit knowledge of the PDE coefficients in the interior, i.e., the FDGQR directly “learns” the problem from its transfer function. This embedding property found applications in PDE solvers, inverse problems and unsupervised machine learning.

Here we show a generalization of this approach to dissipative PDE problems, e.g., electromagnetic and acoustic wave propagation in lossy dispersive media. Potential applications include solution of inverse scattering problems in dispersive media, such as seismic exploration, radars and sonars.

To fix the idea, we consider a passive irreducible SISO ROM

\[ f_n(s) = \sum_{j=1}^{n} \frac{y_i}{s + \sigma_j}, \]  

assuming that all complex terms in (1) come in conjugate pairs.

We will seek ladder realization of (1) as

\[ r_j u_j + v_j - v_{j-1} = -s \tilde{h}_j u_j, \]
\[ u_{j+1} - u_j + \tilde{r}_j v_j = -s h_j v_j, \]

for \( j = 0, \ldots, n \) with boundary conditions

\[ u_{n+1} = 0, \quad v_1 = -1, \]

and \( 4n \) real parameters \( h_i, \tilde{h}_i, r_i \) and \( \tilde{r}_i, \ i = 1, \ldots, n \), that can be considered, respectively, as the equivalent discrete inductances, capacitors and also primary and dual conductors. Alternatively,
they can be viewed as respectively masses, spring stiffness, primary and dual dampers of a mechanical string. Reordering variables would bring (2) into tridiagonal form, so from the spectral measure given by (1) the coefficients of (2) can be obtained via a non-symmetric Lanczos algorithm written in $J$-symmetric form and $f_n(s)$ can be equivalently computed as

$$f_n(s) = u_1.$$ 

The cases considered in the original FDGQR correspond to either (i) real $y$, $\theta$ or (ii) real $y$ and imaginary $\theta$. Both cases are covered by the Stieltjes theorem, that yields in case (i) real positive $h, \hat{h}$ and trivial $r, \hat{r}$, and in case (ii) real positive $h,r$ and trivial $\hat{h},\hat{r}$. This result allows a simple interpretation of (1) as the staggered finite-difference approximation of the underlying PDE problem. For PDEs in more than one variables (including topologically rich data-manifolds), a finite-difference interpretation is obtained via a MIMO extensions in block form.

The main difficulty of extending this approach to general passive problems is that the Stieltjes theory is no longer applicable. Moreover, the tridiagonal realization of a passive ROM transfer function (1) via the ladder network (2) cannot always be obtained in port-Hamiltonian form, i.e., the equivalent primary and dual conductors may change sign.

Embedding of the Stieltjes problems, e.g., the case (i) was done by mapping $h$ and $\hat{h}$ into values of acoustic (or electromagnetic) impedance at grid cells, that required a special coordinate stretching (known as travel time coordinate transform) for continuous problems. Likewise, to circumvent possible non-positivity of conductors for the non-Stieltjes case, we introduce an additional complex $s$-dependent coordinate stretching, vanishing as $s \to \infty$. This stretching applied in the discrete setting induces a diagonal factorization, removes oscillating coefficients, and leads to an accurate embedding for moderate variations of the coefficients of the continuum problems, i.e., it maps discrete coefficients onto the values of their continuum counterparts.

Not only does this embedding yields an approximate linear algebraic algorithm for the solution of the inverse problems for dissipative PDEs, it also leads to new insight into the properties of their ROM realizations.
On the Convergence of Two-Level AMG with aggregation-based coarsening relying on weighted matching

Pasqua D’Ambra, Fabio Durastante, and Salvatore Filippone

1Istituto per le Applicazioni del Calcolo “M. Picone”, Consiglio Nazionale delle Ricerche
f.durastante@na.iac.cnr.it
2Centre for Computational Engineering Sciences, School of Aerospace, Transport and Manufacturing, Cranfield University

Abstract

In this talk we discuss some theoretical results for a two-level algebraic multigrid method (TL-AMG for short); the method is under active development in the context of scientific applications on supercomputing facilities in the framework of the Horizon 2020 project ”EoCoE-II”. The above mentioned TL-AMG employs an aggregation-based coarsening relying on weighted matching and was originally proposed in [1, 2]. We focus here on a convergence analysis making use of some tools in the style of [3, 4].

Consider the following linear system

\[ Au = f, \]

where \( A \in \mathbb{R}^{n \times n} \) is symmetric positive definite (SPD), \( u, f \in \mathbb{R}^n \). A general two-level algebraic multigrid solver for (1) can be written as a stationary linear iterative method

Given \( u^0 \), compute \( u^m = u^{m-1} + B(f - Au^{m-1}) \), \( m = 1, \ldots \)

where the iteration matrix \( B \), acting as an approximate inverse of \( A \), is constructed from the following building blocks:

- a convergent smoother, \( R : \mathbb{R}^n \rightarrow \mathbb{R}^n \),
- a coarse space \( \mathbb{R}^{n_c} \), \( n_c < n \), linked to \( \mathbb{R}^n \) via a prolongation operator \( P : \mathbb{R}^{n_c} \rightarrow \mathbb{R}^n \),
- a coarse space solver, \( B_c : \mathbb{R}^{n_c} \rightarrow \mathbb{R}^{n_c} \).

The construction of the underlying multigrid operator \( B \) starts with the definition of the aggregates \( \{ A_j \subset \mathcal{I} : A_i \cap A_j = \emptyset \text{ if } i \neq j \}_{j=1}^J \) for the row/column indices \( \mathcal{I} = \{1, \ldots, n\} \) identifying the matrix entries. The aggregates are built using a maximum product matching on the weighted graph \( G = (\mathcal{V}, \mathcal{E}) \) associated with the sparse matrix \( \hat{A} \), which in turn is built from \( A \) and a suitable (algebraically smooth) weight vector \( w \) as follows:

\[ (\hat{A})_{i,j} = \hat{a}_{i,j} = 1 - \frac{2a_{i,j}w_iw_j}{a_{i,i}w_i^2 + a_{j,j}w_j^2}. \]

The prolongation matrix \( P \) is then built by using the entries in \( w \) and the aggregates \( \{ A_j = \{ m_k \}_{k=1}^2 \}_{j} \). Specifically, if we denote by \( n_p = |\mathcal{M}| \) the cardinality of the graph matching \( \mathcal{M} \) and by \( n_s \) the number of unmatched edges, we can identify for each edge \( e_{i \rightarrow j} \in \mathcal{M} \) the vectors

\[
\begin{align*}
  w_{e_{i \rightarrow j}} &= \frac{1}{\sqrt{\frac{w_i^2}{a_{i,i}} + \frac{w_j^2}{a_{j,j}}}} \begin{bmatrix} w_i \\ a_{i,i} \end{bmatrix}, \\
  w_{e_{i \rightarrow j}}^\perp &= \frac{1}{\sqrt{\frac{w_i^2}{a_{j,j}} + \frac{w_j^2}{a_{i,i}}}} \begin{bmatrix} -w_j \\ a_{j,j} \end{bmatrix}.
\end{align*}
\]
We define the linear operator $\Pi_j$ for the aggregates $A_j$ as

$$(\Pi_j v)_i = \begin{cases} v_k, & i = m_k, \\ 0, & i \notin A_j \end{cases}$$  \hspace{1cm} (2)$$

and the columns of the corresponding projection matrix are given by

$$\tilde{P} = [p_1, \ldots, p_n], \text{ for } p_j = \Pi_j w_{e_{i,j}},$$

while the remaining columns of the prolongator are

$$W = [p_{n_p+1}, \ldots, p_{n_p+n_s}] = [p_{n_p+1}, \ldots, p_{J}], \text{ for } p_k = \Pi_k \frac{w_k}{|w_k|}.$$  

Thus, the prolongator is $P = [\tilde{P} \ W]$; it defines the $\text{diag}(A) = D$–orthogonal projector

$$Q = P(P^T D P)^{-1}P^T D = PD^{-1}P^T D = \text{diag}(Q_1, \ldots, Q_J),$$

from which we can express the convergence rate of the multigrid algorithm as

$$\|E\|_A = \|I - BA\|_A \leq 1 - \frac{\mu_c}{c_D}.$$  

The constant $c_D$ depends on the convergent smoother, i.e.,

$$c_D \|v\|_2 \leq \|v\|_{R^{-1}} \leq c_D \|v\|_2, \text{ for } R = R + R^T - R^T A R,$$

and $\mu_c$ is the convergence rate for global coarse space $V_c$, defined as

$$V_c = \mathbb{R}^{n_c} = \sum_{j=1}^J \Pi_j V^c_j$$

by means of the local prolongators (2) and the coarse spaces $V^c_j \subset V_j$ associated to the aggregates $A_j$. The analysis is therefore reduced to providing a bound for $\mu_c = \min_{1 \leq j \leq J} \mu_j(V^c_j)$ in terms of the constants

$$\mu_j^{-1}(V^c_j) = \max_{v_j \in V_j} \frac{\|(I - Q_j)v_j\|_{D_j}^2}{\|v_j\|_{A_j}^2}, \text{ diag}(D_1, \ldots, D_j) = P^T D P, \text{ diag } (A_1, \ldots, A_j) = P^T A P.$$  

The convergence theory is complemented in two ways. First, by the adoption of a bootstrap procedure for refining the computation of the algebraically smooth vector $w$; this can be seen as a strategy for boosting the convergence rate of a single multigrid hierarchy [1, 2]. Second, we perform a weak–scalability analysis, demonstrating the applicability of the proposed algorithm in a high–performance computing environment.

References


Automatic differentiation (AD) is fundamental to gradient based optimization of neural networks and is used throughout scientific computing. There are two popular approaches to AD namely forward and reverse modes [1, 2]. A common high level description of AD is that it is really “only” the chain-rule. Recently automatic differentiation is taking on a new life in the form of Differentiable Programming which appears to be breathing new life to the centuries old technology of taking derivatives.

This paper introduces the notion that AD is best understood with a matrix based approach. We argue that a linear algebra based framework for AD, while mathematically equivalent to other approaches, provides a simplicity of understanding, and equally importantly a useful approach to implementation.

Regarding software, while most high level languages allow for matrices whose elements are scalars, the ability to work with matrices whose elements might be operators without major changes to the elementwise software is an intriguing abstraction for software. We discuss a Julia implementation that makes this step particularly mathematically natural.

As is well documented in the preface to the book Graph Algorithms in the Language of Linear Algebra [3] there have been many known benefits to formulate mathematically a graph algorithm in linear algebraic terms. One of the known benefits is cognitive complexity. It is our view that a linear algebraic approach sheds light on how back propagation works. For example, we theoretically connect back propagation to the back substitution method for triangular systems of equations. Similarly, forward substitution corresponds to forward mode calculation of automatic differentiation.

The ability to implement these abstractions and retain performance is demonstrated in Julia, a language that facilitates abstractions, multiple dispatch, and which offers generic operators. Two main contributions are:

- Emphasizing the adjacency matrix representation for neural networks to calculate gradients,
- Showing that forward mode and reverse mode differentiation are two natural ways of solving the linear equation that occurs in the new representation.

In the presented framework, the gradient calculation in a neural network, with parameter $p$, is reduced to Equation (1), where $L$ is a lower triangular matrix that stores the connection “weights” in the computation graph of a neural network, and $D$ is a diagonal matrix that stores partial derivatives of the weights w.r.t each layer of the neural network. In general, the elements of $D$, and $L$ are corresponding linear operators or their underlying matrices.

$$\nabla_p J = D^T(I - L^T)g$$

(1)
For a scalar neural network, where all parameters and inputs are scalar, we can show that Forward Mode differentiation and Reverse Mode differentiation corresponds to specific cases of solving the equation in either order. Let us suppose a neural network has the adjacency matrix in Equation (2)

$$L^T = \begin{pmatrix} 
. & . & a & . \\
. & b & . & . \\
. & c & . & . \\
. & d & . & . 
\end{pmatrix} \tag{2}$$

We recall from elementary linear algebra that we can factor \((I - L^T)^{-1}\) with elimination matrices. For example, for this \((I - L^T)^{-1}\),

\[
(I - L^T)^{-1} = E_1E_2E_3E_4 = \\
\begin{pmatrix}
1 & a & . & . \\
. & 1 & b & . \\
. & 1 & . & c \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
\end{pmatrix}
\begin{pmatrix}
1 & . & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
\end{pmatrix}
\begin{pmatrix}
1 & . & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
\end{pmatrix}
\begin{pmatrix}
1 & . & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
0 \\
1
\end{pmatrix}
\]

It follows that

\[
(I - L^T)^{-1} = \\
\begin{pmatrix}
1 & a & . & . \\
. & 1 & b & . \\
. & 1 & c & . \\
. & 1 & d & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
\end{pmatrix}
\begin{pmatrix}
1 & . & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
\end{pmatrix}
\begin{pmatrix}
1 & . & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
. & 1 & . & . \\
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
0 \\
1
\end{pmatrix}
\]

It is well known [4], that there are a Catalan number, \(C_5 = 42\), ways to parenthesize the above expression. One of which evaluates left to right; this is forward substitution which computes the graph weights forward. Another evaluates from right to left, backward substitution. There are three other “mixed-modes” [5] which combine forward and backward methods. The remaining 37 methods require matrix-matrix multiplication as a first step.

References


Partitioning Large and Sparse Tensors using Best Low Rank Decompositions

Lars Eldén, and Maryam Dehghan

Department of Mathematics, Linköping University
http://users.mai.liu.se/larel04/

Abstract

Extracting information from large, sparse matrix data sets can be done by clustering algorithms. One family of algorithms is based on spectral graph partitioning, where one computes the two smallest eigenvalues and corresponding eigenvectors of the graph Laplacian. A good partitioning of the graph can often be found by grouping the nodes of the graph according to the sign of the entries of the eigenvector corresponding to the second smallest eigenvalue. Similar algorithms can be based on the computation of a few of the largest singular values and corresponding singular vectors of the data matrix [1].

In modern data science applications data organized as tensors are ubiquitous. To our knowledge no natural generalization of the graph Laplacian to nonnegative tensors exists. However, a generalization is possible based on one of the tensor concepts corresponding to the matrix SVD: the best low rank approximation (in the sense of Tucker). Thus we compute best rank-(2,2,2) approximation and use the corresponding vectors to determine a partitioning of the tensor.

We outline an algorithm for the computation of the best low rank approximation of a tensor. It is a generalization of the Krylov-Schur algorithm for matrices.

The algorithms are applied to several problems from data science, with sparse tensors of dimensions of the order $10^4 \times 10^4 \times 10^3$.

References

[1] I. S. Dhillon, Co-Clustering Documents and Words Using Bipartite Spectral Graph Partitioning
Proc. 7th ACM-SIGKDD Conference, 2001
Compact Reduced-Basis Methods for Time-Dependent Partial Differential Equations

Howard C. Elman

Department of Computer Science and
Institute for Advanced Computer Studies
University of Maryland
www.cs.umd.edu/~elman

Abstract

This work concerns efficient methods for computing solutions to discrete parameter-dependent evolutionary partial differential equations

\[ u_t = F_\xi(u), \]  

where \( \xi \) is a vector of (one or more) parameters. The solution \( u(x,t) = u_\xi(x,t) \), which is sought on an interval \([t_0,T]\), is also parameter dependent. Such problems arise when some terms in the problem statement, such as diffusion coefficients or boundary conditions, depend on parameters that are unknown or for which there is interest in solutions for multiple values. If discrete solutions to (1) require fine spatial resolution or multiple time steps, or if solutions are sought for many parameters (for example, in Monte Carlo simulation), then costs may be prohibitive. One way to address this is to develop reduced-order models, whereby approximate solutions \( \hat{u}_\xi \approx u_\xi \) can be obtained from a space spanned by a reduced basis \( \{u_\xi j\}_{j=1}^{n_r} \), where the dimension \( n_r \) of the reduced space is much smaller than \( N_h \), the size of the spatial discretization [3].

Consider for simplicity the linear case \( F_\xi(u) = L_\xi u \). (Nonlinear problems can be handled using empirical interpolation methods [1, 4] Discretization in space then gives a linear system of ordinary differential equations (ODEs) of the form \( M_u = A_\xi u \), which can be treated using standard ODE solvers. Suppose \( \{u_\xi j\}_{j=1}^{n_r} \) is a reduced basis and let \( V \in \mathbb{R}^{N_h \times n_r} \) be a matrix whose range spans the space as that spanned by the basis; \( V \) is typically constructed as an orthogonal matrix. The Galerkin form of the discrete reduced problem is \( V^T M V \dot{y} = V^T A_\xi V y \), where \( \hat{u}_\xi = V \dot{y} \). Reduced-basis methods construct the basis by solving the full-sized discrete system for a specific “training set” of parameters \( \{\xi_j\} \), and then performing the simulation for a larger number of parameters using the reduced problem. The first, “offline” step, may be expensive; it builds the reduced basis by augmenting it with an iterative strategy, using greedy search to ensure that an error indicator \( \eta(\hat{u}_\xi) \) is small for reduced solutions on the training set. The second, “online” step, uses the reduced model to perform the simulation for multiple parameters; since the reduced problem is much smaller, each step is intended to be inexpensive.

This study addresses a complication arising for time-dependent problems. In this case, the character of the solution may vary during the course of time integration, and it may be that a basis suitable at some time step may not be effective at other time steps. There are several ways to handle this. One approach [2] is to fix the time-stepping strategy for all choices of parameters, giving a set of time steps \( \{t_0, t_1, t_2, \ldots, t_{N_t} = T\} \), and then to construct a collection of reduced basis sets \( \{u_j^{(k)}\} \), where the \( k \)th set is to be used on interval \([t_k, t_{k+1}]\). This allows each individual reduced basis to be small, but has the disadvantage of requiring a fixed set of predetermined time steps for both the offline and online computations and thus precludes adaptive time-stepping. An alternative is to allow adaptive time-stepping in the offline computations, and produce a reduced basis defined
on the whole time interval $[t_0, T]$. This would also allow use of adaptive time integration for online computations, but it would also require a significantly larger basis. We explore a compromise between these approaches:

- For some first parameter $\xi_1$, generate a complete set of (full-size) solutions $\{u_{t_j}\}_{j=0}^{N_t}$ across $[t_0, T]$ by a suitable (adaptive) time-stepping strategy.
- Organize the collection of time steps into $n_t$ sets of approximately equal size, $N_t/n_t$, grouped into intervals $I_1, I_2, \ldots, I_{n_t}$, where the $k$th set consists of all time steps in interval $I_k$.
- For each $k = 1, \ldots, n_t$, construct a reduced basis $V_k$ to be used on interval $I_k$.

Here, $n_t = 1$ corresponds to constructing a basis for the whole interval, and the aim is to choose $n_t = O(1)$ but large enough so that the reduced basis size and ensuing online costs are small. The outline above identifies the first step of the construction of the reduced basis. The basis is built using a greedy search on the training set of parameters and augmenting each component basis $V_k$ using the (full-sized) solution for which error indicator is maximal.

We show that this strategy leads to reduced bases of dimension $n_r$ smaller than that built from the full time interval, while allowing adaptive time-stepping for the reduced model, and requiring computational costs that depend only on $n_r$ and independent of $N_h$.

References


Iteratively Corrected Rational Interpolation. for Nonlinear Eigenvalue Problems

Jonathan Baker, Mark Embree, and Serkan Gugercin

Department of Mathematics, Virginia Tech, Blacksburg, VA, 24061, USA
https://intranet.math.vt.edu/people/embree

Abstract

A scalar \( \lambda \in \mathbb{C} \) is an eigenvalue of the analytic matrix-valued function \( T(z) : \mathbb{C} \rightarrow \mathbb{C}^{n \times n} \) if there exists some nonzero \( v \in \mathbb{C}^n \) such that \( T(\lambda)v = 0 \). Common nonlinear \( T(z) \) give infinitely many eigenvalues, and we seek some subset of these eigenvalues in a target region of the complex plane.

For modest dimensions \( n \), algorithms based on complex contour integration or linearizations (see, e.g., [3]) are appealing. For large \( n \), projection methods provide a way to reduce the dimension while preserving the structure of the nonlinearity. We can always write, for some \( m \leq n^2 \),

\[
T(z) = \sum_{j=1}^{m} f_j(z)A_j.
\]

Galerkin projection methods pick some matrix \( U \in \mathbb{C}^{n \times r} \) with orthonormal columns and approximate \( T(z) \) with \( T_r(z) : \mathbb{C} \rightarrow \mathbb{C}^{r \times r} \), where

\[
T_r(z) := \sum_{j=1}^{m} f_j(z)U^*A_jU.
\]

Notice that \( T_r(z) \) has the same nonlinearity as \( T(z) \). (For example, to analyze stability of solutions to the delay differential equation \( x'(t) = Ax(t) + x(t-1) \), one seeks the rightmost eigenvalues of \( T(z) = zI - A - e^{-z}I \); for this problem, \( T_r(z) = zI - U^*AU - e^{-z}I \).) This projection approach differs in spirit from linearization or rationalization methods (see, e.g., [4]), which approximate the nonlinear functions \( f_j \) while keeping the full-order coefficient matrices \( A_j \in \mathbb{C}^{n \times n} \) intact.

How should one choose the subspace \( \text{Range}(U) \)? This is a fundamental question for any projection method. In this talk, we advocate an approach motivated by tangential rational interpolation, which also suggests a method for updating the projection subspace to drive the approximation toward convergence while maintaining a fixed dimension \( r \).

(1) Pick \( r \) interpolation points \( \{z_j\}_{j=1}^{r} \) and interpolation directions \( \{r_j\}_{j=1}^{r} \).

(2) Construct an orthonormal basis for the projection space:

\[
U = \text{orth}([T(z_1)^{-1}r_1, T(z_2)^{-1}r_2, \ldots, T(z_r)^{-1}r_r]).
\]

(3) Form the reduced-dimension nonlinear function

\[
T_r(z) := U^*T(z)U \in \mathbb{C}^{r \times r}.
\]

The projected function \( T_r(z) \) satisfies a key tangential interpolation theorem: \( T_r(z)^{-1} \) interpolates \( T(z)^{-1} \) at the interpolation points \( \{z_j\} \) in the directions \( \{r_j\} \).
Theorem. Suppose no interpolation point is an eigenvalue of $T(z)$ or $T_r(z)$. Then

$$T(z_j)^{-1}r_j = UT_r(z_j)^{-1}U^*r_j, \quad j = 1, \ldots, r.$$  

One can now compute the eigenvalues of the $r$-dimensional function $T_r(\lambda)$ in the region of $\mathbb{C}$ of interest using algorithms based on contour integration or linearization. (Note that since the nonlinear structure is preserved, this projected function could also have infinitely many eigenvalues.) Despite the tangential interpolation property, we do not generally expect the eigenvalues of $T_r(z)$ to be highly accurate. However, we hope they are sufficiently accurate to suggest an improvement to the interpolation points and directions. So motivated, we add an iterative step.

(4) Compute the eigenvalues of $T_r(z)$ and use the resulting eigenvalues and eigenvectors to update $\{z_j\}_{j=1}^r$ and $\{r_j\}_{j=1}^r$. Repeat the algorithm, starting at step (2).

This interpolatory projection approach is motivated by algorithms for model reduction of nonlinear systems with coprime factorizations [1]. The iterative improvement of points and directions is inspired by the dominant pole algorithm [5] and the Iterative Rational Krylov Approximation (IRKA) [2] for interpolatory model reduction.

How should one best update the interpolation points and directions? For example, one can choose all the new points $\{z_j\}_{j=1}^r$ to be eigenvalues of the last $T_r(z)$, or leave some interpolation points fixed in a region of interest in the complex plane. The interpolation directions could be random vectors, eigenvectors of $T_r(z)$, or singular vectors of $T(z_j)$ corresponding to smallest singular values. We will discuss our experience with a variety of choices for these parameters, illustrating their performance with computational examples.

References


Fast Generation of Extreme-Scale Matrices with Pre-Assigned 2-Norm Condition Number

Massimiliano Fasi and Nicholas J. Higham

Department of Mathematics, The University of Manchester
http://www.ma.man.ac.uk/~mbbxmf2

In order to assess experimentally the stability of algorithms for the solution of systems of linear equations, it is typically desirable to have a certain degree of control over the condition number of the test matrices being used. If the tests are being performed at scale (e.g., in an HPL benchmark run for a TOP500 submission), it is necessary to ensure that generating the test data will take up only a negligible portion of the overall execution time required to solve the linear system. We develop two new techniques that satisfy these requirements and can be used to efficiently construct extremely large matrices with preassigned 2-norm condition number. Focusing on distributed memory environments, we discuss how these can be implemented in a communication-avoiding fashion.

Test matrices of moderate size with pre-assigned 2-norm condition number can be easily generated by means of a strategy, originally proposed by Demmel and McKenney [1], which exploits the SVD decomposition together with a technique, due to Stewart [2], for efficiently applying random orthogonal transformations without explicitly generating the underlying orthogonal matrix. It is well known that the \( n \)-by-\( n \) matrix

\[
A := U \Sigma V^T, \quad U, V \in \mathbb{R}^{n \times n} \text{ are orthogonal and } \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^{n \times n}, \quad \sigma_1 \geq \cdots \geq \sigma_n \geq 0,
\]

has 2-norm condition number \( \kappa_2(A) = \sigma_1 \sigma_n^{-1} \) if \( \sigma_n \neq 0 \) and \( \kappa_2(A) = \infty \) otherwise. Therefore, one can simply produce a random orthogonal matrix, scale its rows by suitably chosen constants, and then apply a second random orthogonal transformation to the resulting matrix. Due to its cubic computational cost, asymptotically speaking this procedure can be even more expensive than the factorization-based method used to solve the linear system, and is thus ill-suited to the extreme-scale setting.

This issue can be addressed in several ways. For instance, one could set either \( U \) or \( V \) to be the identity matrix, greatly reducing the cost of the algorithm. This solution, however, is not always adequate, as a matrix generated in this way would be just a row or column scaling of an orthogonal matrix, and some algorithms might be able to exploit this rather special structure. We propose the more refined strategy of picking an arbitrary orthogonal matrix on one side and a Householder transformation on the other. Here we focus, in particular, on the construction of the matrix

\[
A = Q \Sigma (I - \alpha uu^T) = Q \Sigma - \alpha (Q \Sigma u) u^T =: Q \Sigma - \alpha yu^T,
\]

where \( Q \in \mathbb{R}^{n \times n} \) is orthogonal, \( u \in \mathbb{R}^n \) is a non-zero vector, \( \alpha := 2\|u\|_2^2 \), and \( \Sigma \) is as in (1); but analogous considerations can be made for the matrix \( (I - \alpha uu^T) \Sigma Q \), or for the more general case of rectangular matrices with pre-assigned singular values.

If each of the elements of \( Q \) can be computed in constant time, the overall cost of the algorithm underlying (2) is quadratic. As we focus on a distributed memory setting, the number of floating point operations alone may not be a good measure of the global performance of the algorithm. Our experimental results, however, show that the new technique is drastically faster than the method proposed by Demmel and McKenney, as the communication-heavy matrix-matrix multiplication is replaced by two lighter matrix operations: a matrix-vector multiplication and an outer product.
A careful choice of $u$, $\Sigma$, and $Q$ can further reduce the computational cost of the algorithm, by completely eliminating, at the price of a marginal increase in the computational cost, the need of communication among the processes involved in the computation. If a symmetric orthogonal matrix $Q$ is picked, and $u$ is chosen to be the $\ell$th row of $Q$, then one can show that the $i$th entry of the vector $y$ in (2) can be written as

$$y_i = \sum_{k=1}^{m} q_{ki} \sigma_k q_{k\ell} + \delta_{i\ell} - \sum_{k=1}^{m} q_{ki} q_{k\ell} = \sum_{k=1}^{m} q_{ki} (\sigma_k - 1) q_{k\ell} + \delta_{i\ell},$$

(3)

where $\delta_{i\ell}$ denotes the Kronecker delta, which equals 1 if $i = \ell$ and 0 otherwise. Equation (3) shows that the number of flops required to compute $y$ can be arbitrarily reduced by setting as many entries of $\sigma$ as possible to 1. Since in our application we are interested only in the ratio of the largest to the smallest singular value, a reasonable choice is to set $\sigma_k = 1$ for $k$ between 2 and $m-1$. Further numerical considerations suggest that it is appropriate to take $\sigma_1 = \sqrt{\kappa}$ and $\sigma_m = \sqrt{\kappa^{-1}}$, so that (3) reduces to

$$y_i = q_{1i} \left( \sqrt{\kappa} - 1 \right) q_{1\ell} + q_{mi} \left( \sqrt{\kappa^{-1}} - 1 \right) q_{m\ell} + \delta_{i\ell},$$

which shows that each of the entries of $y$ can be computed with a constant number of floating point operations.

In order to see how the algorithm can be implemented in a communication-avoiding fashion, we note that to generate the element $a_{ij} = \sigma_j q_{ij} - 2 y_i q_{ij} a_{ij}$ a process needs only the corresponding element of $Q$, the entries of $y$ and $q_{ij}$ matching its row and column indices, respectively, and the desired 2-norm condition number $\kappa$. If the matrix is distributed by blocks, as it is the case, for instance, for ScaLAPACK’s two-dimensional block-cyclic distribution, then each processor can efficiently generate a block without any communication at the price of the (linear) extra cost of computing locally the portions of $y$ and $q_{ij}$ that are needed. We note that these algorithms are deterministic, in the sense that they will always produce the same test matrix once the orthogonal matrix $Q$, the index $\ell$, and the singular values $\sigma_1$ and $\sigma_n$ have been chosen.

We have implemented the new algorithms using the BLACS, BLAS, PBLAS, and ScaLAPACK libraries, and have tested them on a distributed memory cluster using up to 1024 cores. Our experimental results show that the performance of our codes makes the new approach a viable strategy for generating extremely large matrices in supercomputers. The new algorithms can be used to produce matrices with pre-assigned 2-norm condition number on single-core machines or in shared-memory environments. Also in these cases the new algorithms are several order of magnitudes faster than the approach by Demmel and McKenney. This is not surprising, since the computational cost of the former scales quadratically, whereas the number of floating point operations required by the latter grows cubically with the order of the matrix being constructed.

As the main application of these methods will be the creation of test sets for experimenting with numerical algorithms at scale, we have also studied the numerical properties of the matrices they generate, finding that they exhibit a reasonably low growth factor for LU factorization with partial pivoting, and typically have an $\infty$-norm condition number just larger than the 2-norm condition number that is used to construct them.
References


Higher-order Markov Chains, Tensor Equations, and Ergodicity Coefficients

Dario Fasino and Francesco Tudisco

Dept. Mathematics, Computer Sci. and Physics, University of Udine, Italy
http://people.uniud.it/page/dario.fasino

Abstract

Random walks are fundamental tools to simulate diffusion processes, define centrality measures and solve community detection problems in complex networks. A random walk is a discrete-time Markov chain described by the iteration $x_{k+1} = Px_k$ where $P$ is the matrix whose entry $P_{ij}$ represent the probability of the state transition $j \rightarrow i$. The matrix $P$ is stochastic, i.e., $\sum_i P_{ij} = 1$. The limiting point of that iteration is a Perron eigenvector of $P$ and represents the stationary distribution of the Markov chain, that is, the long-term fraction of the time spent in each state during the process. Relevant questions concerning the convergence speed of the process, the uniqueness of the limiting distribution and its sensitivity to perturbations can be answered by means of the so-called ergodicity coefficients, without recourse to spectral properties of $P$. These coefficients are designed to estimate the rate at which the powers $P^k$ converge to a rank-one matrix, and can be computed by means of explicit formulas in the entries of $P$ [4]. Probably, the most famous ergodicity coefficient is the Dobrushin coefficient,

$$\tau(P) = \frac{1}{2} \max_{j,k} \sum_i |P_{ij} - P_{ik}|,$$

which shows up in the following well known result.

**Theorem.** If $\tau(P) < 1$ then there exists a unique probability vector solving the eigenvector equation $x = Px$. Moreover, the iteration $x_{k+1} = Px_k$ converges to $x$ for every initial probability vector $x_0$, and $\|x_k - x\|_1 \leq \tau(P)^k \|x_0 - x\|_1$.

A natural extension of Markov chains is to have the state transitions depend on the past few states, rather than just the last one. These processes are called higher-order Markov chains and are much better at modeling data in a variety of applications, where the additional memory of the past offers a better descriptive and predictive value. For example, a second order Markov chain is defined by a third order stochastic tensor (i.e., hypermatrix) $P = (P_{ijk})$ where $P_{ijk}$ is the probability of transitioning to state $i$, given that the last state was $j$ and the second last state was $k$, with $\sum_i P_{ijk} = 1$. Using nowadays standard notations, that process can be written as $x_{k+1} = Px_k x_{k-1}$ and the respective limiting vector fulfills the quadratic vector equation $x = Px x$, that is, it is the stochastic Perron Z-eigenvector of $P$.

Vertex reinforced random walks are another important example of higher-order stochastic processes. Here, the state transition probabilities at each time step depend on the whole history of the process. For instance, the spacey random walk by Benson, Gleich and Lim [1] belongs to the general scheme

$$x_{k+1} = Px_k y_k, \quad y_{k+1} = c_k x_k + (1 - c_k) y_k,$$

where $P$ is a stochastic tensor and $c_k \in (0, 1)$, for $k = 1, 2 \ldots$ In some sense, these processes can be looked at as Markov chains where the transition matrix at time $k$ depends on the auxiliary vector $y_k$, which records the past history of the process.

However, the analysis of nonlinear and higher-order stochastic processes is more complicated than what we could expect, even if they are defined in terms of the action of multilinear maps, that is,
tensors. In fact, unlike in the matrix case, the positivity of the relevant operators fails to assure
the uniqueness of fixed points. Moreover, the positivity plus the uniqueness of fixed points does
not imply the convergence of the process to the stationary density, nor the contractivity of the map
defining the associated dynamical system.

By extending the wide literature on ergodicity coefficients for matrices, we introduce a family of
higher-order ergodicity coefficients for three-mode tensors and discuss how these allow to derive new
conditions on the existence, uniqueness and computability of stationary distributions for higher-
order Markov models described by stochastic tensors [3]. An immediate example of an ergodic
coefficient for third order tensors is the Dobrushin-type coefficient

$$\mathcal{T}(P) = \frac{1}{2} \max_{j,k_1,k_2} \sum_i |P_{ijk_1} - P_{ijk_2} + P_{ik_1j} - P_{ik_2j}|,$$

which is the best Lipschitz constant (over the set of probability vectors) of the quadratic map
$$x \mapsto Pxx$$ with respect to the 1-norm. The preceding definition helps in obtaining the following
convergence result for a widespread nonlinear stochastic process.

**Theorem.** If $\mathcal{T}(P) < 1$ then there exists a unique stochastic solution of the $Z$-eigenvector equation
$$x = Pxx.$$ Moreover, the iteration $x_{k+1} = Px_k x_k$ converges to $x$ for every initial probability vector
$x_0$, and $\|x_k - x\|_1 \leq \mathcal{T}(P)^k \|x_0 - x\|_1$.

In this work, we show that higher-order ergodicity coefficients allow to easily derive uniqueness
results for the stationary distribution of various nonlinear and higher-order stochastic processes,
besides those mentioned above, under hypotheses that are quite simple to test and usually weaker
than those in analogous known results. Moreover, we focus on the coupled tensor (multivariate
polynomial) equations

$$P_{xy} = x, \quad Q_{xx} = y,$$

where $P$ and $Q$ are two given stochastic tensors, and on the modified version

$$(1 - \alpha)P_{xy} + \alpha v = x, \quad (1 - \beta)Q_{xx} + \beta w = y,$$

where $v$ and $w$ are two assigned probability vectors and $0 < \alpha, \beta < 1$ are PageRank-style parameters
that control their influence. These equations are prototypical of a number of Information Retrieval
models to evaluate the relevance of nodes and edges in complex networks, including the multilinear
PageRank [2]. In fact, the solutions of the previous equations correspond to stationary densities of
stochastic processes modeling random surfers on networks encoding multi-dimensional and multi-
relational data [5, 6].

**References**


Inexact Methods for the Low-Rank Solution to Large-Scale Lyapunov Equations

Melina A. Freitag, and Patrick Kürschner

Institut für Mathematik, Universität Potsdam

www.math.uni-potsdam.de/professuren/datenassimilation/personen/prof-dr-melina-freitag/

Abstract

We consider the numerical solution of large scale Lyapunov equations of the form

$$AX + XA^T = -BB^T,$$

where $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times r}$. Lyapunov equations play a fundamental role in many areas of applications, such as control theory, model reduction and signal processing, see, e.g. [1]. Here we assume that the spectrum of $A$, $\Lambda(A)$, lies in the open left half plane, and that the right hand side is of low rank, e.g. $r \ll n$, which is often the case in practice. A large and growing amount of literature considers the solution to (1), see [2] and references therein for an overview of developments and methods.

The solution matrix $X$ of (1) is, in general, dense, making it virtually impossible to store it for large dimensions $n$. For a low-rank right hand side, however, the solution $X$ often has a very small numerical rank, see for example [3, 4, 5, 6] and many algorithms have been developed that approximate $X$ by a low-rank matrix $X \approx ZZ^T$, where $Z \in \mathbb{R}^{n \times s}$, $s \ll n$. Important low-rank algorithms are, for instance, projection type methods based on Krylov subspaces [7, 8, 9, 10, 11] and low-rank alternating directions implicit (ADI) methods [3, 12, 13, 14, 15, 16]. Here, we consider both the rational Krylov subspace method (from the family of projection type methods) and the low-rank ADI method. One of the computationally most expensive parts in both methods is that, in each iteration step, shifted linear systems of the form

$$(A - \sigma I)y = z, \quad z \in \mathbb{R}^{n \times r},$$

need to be solved, where the shift $\sigma$ is usually variable and both the shift $\sigma$ and the right hand side $z$ depend on the particular algorithm used. Normally these linear systems are solved by sparse-direct or iterative methods. When iterative methods, such as preconditioned Krylov subspace methods, are used to solve the linear systems, then these solves are implemented inexactly and we obtain a so-called inner-outer iterative method (sometimes the term “inexact method” is used). The outer method is (in our case) a rational Krylov subspace method or a low-rank ADI iteration. The inner problem is the iterative solution to the linear systems. The inner solves are often carried out at least as accurately as the required solution accuracy for the Lyapunov equation (cf., e.g., the numerical experiments in [10]), usually in terms of the associated Lyapunov residual norms. It turns out that this is not necessary, and the underlying theory is the main contribution of this work.

Inexact methods have been considered for the solution to linear systems and eigenvalue problems (see [17, 18, 19, 20] and references therein). One focus has been on inexact inverse iteration and similar methods, where, in general, the accuracy of the inexact solve has to be increased to obtain convergence [21]. For subspace expansion methods, such as the Krylov methods considered in [18, 17, 19], it has been observed numerically and shown theoretically that it is possible to relax the solve tolerance as the outer iteration proceeds.
We show that for both the rational Krylov subspace method and the low-rank ADI algorithm we can relax the stopping tolerance within the inner solve, a similar behavior as observed for Krylov methods with inexact matrix-vector products applied to eigenvalue methods [19, 20], linear systems [17, 18], and matrix functions [22]. We provide practical relaxation strategies for both the rational Krylov subspace method and the low-rank ADI algorithm and present numerical examples which support the theory. The presentation is based on the work in [23].

References


Krylov type methods exploiting the quadratic numerical range

Andreas Frommer, Birgit Jacob, Karsten Kahl, Christian Wyss and Ian Zwaan

Department of Mathematics, University of Wuppertal
www.fk4.uni-wuppertal.de

Abstract

Assume that \( A \in \mathbb{C}^{n \times n} \) has a “natural” block decomposition of the form
\[
A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}
\]
with \( A_{ij} \in \mathbb{C}^{n_i \times n_j}, i,j = 1,2, \) \( n_1 + n_2 = n. \) (1)

All vectors \( x \) from \( \mathbb{C}^n \) are endowed with the same block structure
\[ x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad x_i \in \mathbb{C}^{n_i}, i = 1,2. \]

The definition of the quadratic numerical range goes back to [1], where it was introduced as a tool to localize spectra of block operators in Hilbert space.

**Definition.** The quadratic numerical range \( W^2 \) of \( A \) is given as
\[
W^2(A) = \bigcup_{\|x_1\| = \|x_2\| = 1} \text{spec} \left( \begin{bmatrix} x_1^* A_{11} x_1 & x_1^* A_{12} x_2 \\ x_2^* A_{21} x_1 & x_2^* A_{22} x_2 \end{bmatrix} \right).
\]

The following basic properties are, e.g., proved in [2]

(i) \( W^2(A) \) is compact,

(ii) \( W^2(A) \) has at most two connected components,

(iii) \( \text{spec}(A) \subseteq W^2(A) \subseteq W(A), \)

(iv) \( W(A_{11}), W(A_{22}) \subseteq W^2(A). \)

In particular, if the quadratic numerical range does not contain 0, the matrix \( A \) is nonsingular.

It is well known that standard Krylov subspace methods for solving a system
\[ Ax = b \]
can expose convergence problems, if \( 0 \in W(A) \), the standard numerical range of \( A \). For example, the FOM iterate may not exist in such cases and the GMRES iteration can suffer from stagnation.

Our approach is now to consider variants of the FOM and GMRES iteration in the case that \( 0 \in W(A) \) and \( 0 \notin W^2(A) \). The basic idea is to modify the standard Krylov subspaces in such a way that the projected operator
\[ H_k = V_k^* A V_k, \quad V_k \text{ a modification of the standard Arnoldi basis vector matrix} \]
satisfies
\[ W^2(H_k) \subseteq W^2(A). \]

So, if \( 0 \notin W^2(A) \), we also have \( 0 \notin W^2(H_k) \). This results in a much smoother convergence of the modified FOM method and avoids stagnation of the modified GMRES method.

We will expose elements of a convergence theory of these modified methods and illustrate their efficiency as compared to standard FOM and GMRES for several examples for which we know \( 0 \notin W^2(A) \).
References


Multigrid Methods for Block-Toeplitz linear systems: Convergence Analysis and Applications

Marco Donatelli, Paola Ferrari, Isabella Furci, and Stefano Serra–Capizzano

Department of Mathematics and Informatics, Wuppertal University


Abstract

Among the iterative methods available to solve linear systems where the size \( n \) of the problem is large, multigrid methods represent a valid choice as fast solvers in the case of structured coefficient matrices. The convergence analysis of two-grid and V-cycle methods has been obtained in a compact and elegant form in the case of structured matrix sequences associated with a (possibly multivariate) scalar-valued function \( f \). In the case where the entries are small generic matrices instead of scalars, the general interest mainly regards specific applications, that is, the focus concerns algorithmic proposals with very marginal attention to theoretical results. Indeed, when the matrix-valued symbol is not diagonal, there is still a substantial lack of an effective projection proposal and of a rigorous convergence analysis.

The first and main aim of the talk is to generalize the existing convergence results in scalar settings [1] to systems with coefficient matrix \( A_n(f) \) in the circulant algebra associated with a matrix-valued symbol \( f \). We prove that the optimal convergence rate is independent from the matrix size \( n \) in the case of positive definite block matrices with generic blocks. We exploit the algebra structure of circulant matrices for the theoretical analysis of the two-grid and V-cycle algorithms, and we consider Toeplitz matrices for practical applications.

First we give an overview on the two-grid method, in particular we recall the results concerning the convergence and the optimality of the method for scalar-valued trigonometric polynomial symbols [1]. According to the relevant literature, the classical Ruge and Stüben convergence analysis in [6] is applied in order to split the two-grid convergence proof in the validation of both a smoothing and an approximation condition. The first is easily generalizable in the block setting and we show how it mainly regards the choice of the specific relaxation parameter for the selected iterative method.

Mimicking the proof for the approximation condition from the scalar structures is nontrivial, owing to the noncommutativity of the involved matrix-valued symbols. Then, we mainly focus on the crucial choice of conditions on the trigonometric polynomial used to construct the projector which ensures the optimality of the two-grid method [3].

The main result regards the proof of the approximation property which provides a generalization of the two sufficient conditions present in the scalar setting and it requires a further commutativity condition on the matrix-valued symbol \( p \) of the grid transfer operator.

Furthermore, we define a class of grid transfer operators

\[
p_{n(j)}^{k(j)}(z) = A_{n(j)}(p_z)(K_{n(j)}^T \otimes I_d),
\]

where \( p_z \) is the related \( d \times d \) matrix-valued generating function of \( A_{n(j)}(p_z) \), \( K_{n(j)} \) is a proper \( k(j) \times n(j) \) down sampling matrix associated with the \( j \)–th level, and \( I_d \) the \( d \times d \) identity matrix. We show that, varying \( z \), the class of matrix-valued symbols \( p_z \) satisfies the obtained theoretical conditions and we propose a strategy to insure fast multigrid convergence even for more than two grids.
Moreover, taking inspiration from the approach in [2], we propose a measure of the ill-conditioning of the symbol at the coarser levels in order to choose a robust grid transfer operator and we extend the optimality results to V-cycle.

To show the numerical validity of our theoretical results, we consider the case of large positive definite block linear systems stemming from quadrilateral $Q_{\text{deg}}$ Lagrangian finite element methods (FEM) applied to the Poisson problem and from staggered discontinuous Galerkin (DG) methods for the incompressible Navier-Stokes equations. In these applications, an important step for the numerical approximation involves the solution of linear systems which possess a natural block (and multilevel block) Toeplitz structure, up to a low rank correction due to boundary conditions. Consequently, we test the applicability of our multigrid procedure to the linear systems stemming from the aforementioned approximation of differential operators in the most generic block multilevel setting. Furthermore, we numerically study the best choice of the parameter $z$ in (1) of grid transfer operators such that the conditioning of the symbol [3] at the coarser levels does not worsen.

The resulting numerical tests confirm the theoretical analysis, proving an optimal convergence rate also in the multivariate setting, validating the numerical results also present in [4]. Finally, we conclude observing that our theoretical results can be useful to mathematically support the projection strategies proposed in several applications. For example, the choice of the projector for tensor rectangular FEM $Q_{\text{deg}}$ approximations of any dimension $k$ based on a geometric approach [5], and possible projectors that are often used when FEM problems are solved using multigrid.

References


Balanced truncation for linear switched systems

I.V. Gosea, M. Petreczky, C. Fiter, and Athanasios C. Antoulas

Data-Driven System Reduction and Identification Group, Max Planck Institute, Magdeburg, Germany

Abstract

Model order reduction (MOR) is a methodology used to approximate large complex models of dynamical systems with simpler and smaller models that are able to capture the original dominant characteristics. Such reduced-order models (ROM) are often used as efficient surrogates for the original system in order to reduce the computational complexity of numerical simulations. For an overview of MOR methods, we refer the reader to [1].

A switched system is a dynamical system that consists of a finite number of subsystems and a logical rule that orchestrates switching between these subsystems. The dynamics of the subsystems, or discrete modes, is described by a collection of differential or difference equations. The discrete events interacting with the subsystems are governed by a piecewise continuous function called the switching signal. Switched systems have applications in control of mechanical and aeronautical systems, power converters and in the automotive industry.

A continuous-time linear switched system (LSS) is a control system of the form:

\[
\Sigma : \begin{cases}
\dot{x}(t) = A_{\sigma(t)}x(t) + B_{\sigma(t)}u(t), \\
y(t) = C_{\sigma(t)}x(t),
\end{cases}
\]

(1)

where \( \Omega = \{1, 2, \ldots, D\} \), \( D > 1 \), is a set of discrete modes, \( \sigma(t) \) is the switching signal, \( u \) is the control input, \( x \) is the state, and \( y \) is the observed output.

For \( q \in \Omega \), note that the system matrices \( A_q \in \mathbb{R}^{n_q \times n_q} \), \( B_q \in \mathbb{R}^{n_q \times m} \), \( C_q \in \mathbb{R}^{p \times n_q} \) correspond to the linear subsystem \( \Sigma_q \) active in mode \( q \in \Omega \):

\[
\Sigma_q : \dot{x}(t) = A_q x(t) + B_q u(t), \quad y(t) = C_q x(t).
\]

(2)

The switching signal \( \sigma(t) : [0, T) \to \Omega \) is a piecewise-continuous function defined as \( (T_k := T) \)

\[
\sigma(t) = \begin{cases}
q_1 & \text{if } t \in [0, T_1], \\
q_i & \text{if } t \in (T_{i-1}, T_i), \quad 2 \leq i \leq k.
\end{cases}
\]

(3)

Originally, assume \( x(0) = 0 \). Then, the initial condition of every subsystem depends on the previous dynamics. More precisely, the transition from one mode \( (q_i) \) to the next \( (q_{i+1}) \) is made via the so-called switching or coupling matrices \( K_{q_i,q_{i+1}} \in \mathbb{R}^{n_{q_{i+1}} \times n_{q_i}} \) where \( q_i, q_{i+1} \in \Omega \) as

\[
x(T_i) = K_{q_i,q_{i+1}} \lim_{t \to T_i} x(t).
\]

(4)

In this contribution, we present a model order reduction algorithm for LSSs based on a prolific technique in the literature, i.e. balanced truncation (BT). The main idea behind BT is to transform a dynamical system to a balanced form defined in such a way that appropriately chosen controllability and observability Gramians are equal and diagonal. Then, a reduced-order model is computed by truncating the states corresponding to the small entries of the diagonal Gramians (see [1]).
For LSSs, it may happen that some state components are difficult to reach and observe in some of the modes yet easy to reach and observe in others. In that case, deciding how to truncate the state variables and obtain a meaningful ROM is not a trivial task.

Nevertheless, over the years, BT was extended for reducing LSSs. First, we mention the contributions in [8, 5]. There, it was assumed that all subsystems in (2) can be simultaneously balanced via global Gramian matrices. The computation of these matrices usually relies on solving linear matrix inequalities (LMIs) which is not feasible, at least in a large-scale setup. More recently, BT methods that avoid solving such LMIs have been proposed in [2, 6, 7, 4]. For these contributions, one needs to solve (generalized) Lyapunov equations instead. Additionally, a data-driven MOR technique was also extended to the LSS case in [3], i.e. the Loewner framework.

In this contribution, we propose the definition of new type of Gramians for LSS (originally introduced in [2]). For example, in the case with two switching modes, i.e. \( D = 2 \), the infinite reachability Gramians are solutions of the following system of coupled generalized Lyapunov equations:

\[
\begin{align*}
A_1 P_1 + P_1 A_1^T + K_{2,1} P_2 K_{1,1}^T + B_1 B_1^T &= 0, \\
A_2 P_2 + P_2 A_2^T + K_{1,2} P_1 K_{2,2}^T + B_2 B_2^T &= 0.
\end{align*}
\]

(5)

Similarly, the infinite observability Gramians satisfy the following system of equations:

\[
\begin{align*}
A_1^T Q_1 + Q_1 A_1 + K_{1,2}^T Q_2 K_{1,2} + C_1^T C_1 &= 0, \\
A_2^T Q_2 + Q_2 A_2 + K_{2,1}^T Q_1 K_{2,1} + C_2^T C_2 &= 0.
\end{align*}
\]

(6)

It is to be noted that certain assumptions need to be made in order that the above equations have solutions; these include stability of matrices \( A_q \) and an upper bound for the norm of matrices \( K_{q_1,q_2} \) (see Proposition 3 in [2]). Additionally, one need not explicitly compute the solutions of equations (5) and (6), but only low-rank factors thereof. Afterwards, the proposed method is based on the typical steps of BT. A reduced-order LSS \( \hat{\Sigma} \) is computed as described in Procedure 5.1 from [2].

Furthermore, we present an error bound for the \( L_2 \) norm of the difference between the outputs of the original LSS, i.e., \( y(t) \) and of the reduced LSS, i.e., \( \hat{y}(t) \). This error bound is formulated in terms of singular values of balanced Gramians and it is valid for slow enough switching signals, i.e., signals with large enough dwell time (the minimal amount of time spent in each mode).

**Theorem 1** [2]. Let \( \Sigma \) be the original LSS and let \( \hat{\Sigma} \) be a reduced-order LSS that approximate \( \Sigma \). Then, there exists \( \mu > 0 \) such that for any switching signal \( \sigma \) with minimal dwell time \( \mu \) and for any control input \( u \in L_2(\mathbb{R}^m) \), the following inequality holds:

\[
\|y - \hat{y}\|_2 \leq 2\beta\|u\|_2.
\]

(7)

In (7), the positive real scalar \( \beta \) is written in terms of the neglected singular values of the balanced diagonal Gramians. Additionally, the topics of stability preservation and extension to the general case with \( D > 2 \) subsystems will also be covered in this contribution.

Finally, if time permits, we will briefly address the case of LSS with constrained switching, i.e. where the MOR procedure is tailored to certain switching scenarios dictated by specific applications, as described in [4].

**References**


Automatic Generation of Minimal and Reduced Models for Structured Parametric Dynamical Systems

Peter Benner, Pawan Goyal, and Igor Pontes Duff

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany
https://goyalpike.github.io/

Abstract

Reduced-order modeling of large-scale complex dynamical systems is a necessary step for faster simulation, optimization, control and uncertainty quantification studies. Model-order reduction (MOR) is a prevalent technique, allowing us to determine a reduced-order or low-dimensional representation of high-dimensional parametric models. Developing efficient and robust MOR techniques has been an active research area in the last couple of decades within computational science and engineering. In this paper, we discuss a novel model reduction framework for generalized linear systems, having the transfer functions of the form:

\[ H(s, p) = C(s, p)K(s, p)^{-1}B(s, p), \]

(1)

where

\[ C(s, p) = \sum_{i=1}^{k} \gamma_i(s, p)I, \quad K(s, p) = \sum_{i=1}^{l} \kappa_i(s, p)I, \quad B(s, p) = \sum_{i=1}^{m} \beta_i(s, p)I, \]

(2)

in which \( i \in \mathbb{R}^{n \times n}, B_i \in \mathbb{R}^{n \times m}, \gamma_i(s, p) \) are constant matrices, \( s \) takes values on the imaginary axis, and \( p = [p^{(1)}, \ldots, p^{(d)}] \in \Omega^d \) are the system parameters. \( \kappa_i(s, p), \beta_i(s, p) \) and \( \gamma_i(s, p) \) are smooth functions of \( s \in \mathbb{C} \) and \( p \in \Omega^d \). Additionally, the restrictions \( \kappa_i(\cdot, p), \beta_i(\cdot, p) \) and \( \gamma_i(\cdot, p) \) are assumed to be meromorphic functions. In this paper, we also assume that \( H(s, p) \) is a strictly proper function for all parameters, i.e., \( \lim_{s \rightarrow \pm \infty} H(s, p) = 0 \). The system (1) covers a large class of linear systems, arising in various science and engineering applications, e.g., classical linear systems, second-order systems, time-delay systems, integro-differential systems, and their parameter-dependent variants.

We aim at constructing low-order surrogate models, having a similar structure using Petrov-Galerkin projection as follows:

\[ \hat{H}(s, p) = \hat{C}(s, p)\hat{K}(s, p)^{-1}\hat{B}(s, p), \]

(3)

where

\[ \hat{C}(s, p) = C(s, p)V, \quad \hat{K}(s, p) = W^T K(s, p)V, \quad \hat{B}(s, p) = W^T B(s, p)V. \]

(4)

and \( V \) and \( W \) are the full rank tall and skinny matrices, respectively. Obviously, the quality of the reduced-order models highly depends on the choice of the matrices \( V \) and \( W \). The two most common MOR techniques, namely balanced truncation and interpolation-based methods have been studied in [1, 2, 3]. These techniques are very well studied for standard linear time-invariant systems. However, for the generalized systems, these methods are not well enough developed as of yet.

In this talk, we discuss a novel model reduction framework for generalized linear systems. To that end, we first discuss the connection between interpolation-based MOR methods with the reach-
able and observable subspaces of linear structured parametric systems. We show that if enough interpolation points are taken, then matrices $V$ and $W$ can constructed in such a way that they encode these subspaces. As a consequence, we propose an approach to construct reduced-order systems preserving the common subspaces containing the most reachable as well as the most observable states. This approach can be seen as a combination of the interpolation-based method in [3, 4] and some inspiration from the Loewner framework for first-order systems [5]. Hence, we obtain a minimal-order surrogate model, capturing the dynamics of high-dimensional models with a very high accuracy. Moreover, we study an extension of the proposed methodology to nonlinear systems as well. Furthermore, we pay particular attention to the computational aspects of the proposed approach. We discuss how low-rank solutions of matrix equations can be utilized to speed-up the computations in a large-scale setting. Lastly, we illustrate the efficiency of the proposed methods by means of several examples, arising in science and engineering and show a comparison with the existing approaches.

These results are available in [4, 6].

References


Parallel Variants of the Conjugate Gradient Algorithm  
and their Performance in Finite Precision Arithmetic

Anne Greenbaum, Hexuan Liu, and Tyler Chen

Applied Math Department, University of Washington
http://faculty.washington.edu/greenbau/

Abstract

The conjugate gradient (CG) and Lanczos algorithms are widely used methods for solving Hermitian positive definite linear systems, computing eigenvalues of Hermitian matrices, and, more generally, for computing the product of a function of a Hermitian matrix with a given vector. It is well-known that the methods may behave differently in finite precision arithmetic than they would in exact arithmetic, which means that whenever different (mathematically equivalent) variants are considered – such as variants that make better use of parallelism – one must be careful about the effects on actual performance and accuracy. I will review the assumptions required for the type of error analysis carried out by Paige [3] and others in the 1980’s and 1990’s and discuss several implementations and whether or not they satisfy these assumptions. I will demonstrate that for certain problems, there is a strong correlation between how well these assumptions are satisfied and how well the different variants perform in terms of number of iterations, but for other problems this is not the case.

The two questions to be addressed, when a variant of the CG algorithm is used to solve a symmetric positive definite linear system $Ax = b$, are: How accurate a solution is ultimately attained, and what is the rate of convergence before this final level of accuracy is reached. Since coefficient formulas are based on an updated “residual” vector $r_k$, once this vector starts to differ significantly from the true residual $b - Ax_k$, the computation is unlikely to make more progress. Thus, it is important that roundoff not cause these two vectors to differ greatly before the needed level of accuracy is achieved. For some discussion of the final accuracy level, see, for instance, [1].

We will concentrate more on the rate of convergence before the true and updated residual vectors start to differ significantly. Even during this stage, different variants may show very different behavior on difficult problems. Equally interesting is the fact that on more standard problems, even after agreement with exact arithmetic is lost, the different variants may behave very similarly. An explanation is given in terms of the analogy established in [2] between slightly perturbed CG/Lanczos recurrences and exact CG applied to a matrix $\hat{A}$ with many eigenvalues distributed throughout small intervals about the eigenvalues of the given matrix $A$, the size of the intervals being a function of the size of the perturbations. If a particular CG implementation results in larger perturbations to the recurrence, then one can expect larger interval sizes for the eigenvalues of $\hat{A}$, with this analogy breaking down when the interval size becomes so large that $\hat{A}$ is indefinite. This usually occurs near the point where the true and updated residual vectors in the finite precision CG computation start to differ significantly and the final accuracy level is reached. Before that point, if the size of the intervals containing the eigenvalues of $\hat{A}$, say, $10^{-14}$ or $10^{-7}$, makes a large difference in the behavior of exact CG, then one can expect different CG variants, implemented in double precision arithmetic, to behave very differently. If this is not the case, then the behavior may be very similar. We illustrate these facts using a set of test problems. When eigenvalues of $A$ are fairly uniformly distributed, the different variants that we consider perform similarly in terms of number of iterations, until their final level of accuracy is reached, although that final accuracy level is different for the different variants. When eigenvalues of $A$ are tightly clustered at the low
end and highly spread out at the upper end, not only do the different variants achieve different levels of accuracy, but their behavior before this point is significantly different as well.

A writeup of some of this work can be found at https://arxiv.org/pdf/1905.05874.pdf.

References


A Scalable Approximate Inverse Block Preconditioner for an Incompressible Magnetohydrodynamics Model Problem

Michael Wathen and Chen Greif

Department of Computer Science, The University of British Columbia
http://www.cs.ubc.ca/~greif

Abstract

Incompressible magnetohydrodynamics (MHD) describes the flow of an electrically conductive fluid in the presence of a magnetic field. Given a sufficiently smooth domain \( \Omega \), consider the steady-state incompressible MHD model [3, Chapter 2]:

\[
-\nu \Delta u + (u \cdot \nabla) u + \nabla p - \kappa (\nabla \times b) \times b = f \quad \text{in } \Omega,
\]

\[
\nabla \cdot u = 0 \quad \text{in } \Omega,
\]

\[
\kappa \nu_m \nabla \times (\nabla \times b) + \nabla r - \kappa \nabla \times (u \times b) = g \quad \text{in } \Omega,
\]

\[
\nabla \cdot b = 0 \quad \text{in } \Omega.
\]

Here \( u \) is the velocity, \( p \) is the hydrodynamic pressure, \( b \) is a magnetic field, and the Lagrange multiplier associated with the divergence constraint on the magnetic field is denoted by \( r \). The vector functions \( f \) and \( g \) represent external forcing terms. The three dimensionless parameters that characterize this model are: the hydrodynamic viscosity \( \nu \), the magnetic viscosity \( \nu_m \), and the coupling number \( \kappa \).

To complete the model, we consider the inhomogeneous Dirichlet boundary conditions:

\[
\begin{align*}
\mathbf{u} &= \mathbf{u}_0 \quad &\text{on } \partial \Omega, \\
\mathbf{n} \times \mathbf{b} &= \mathbf{n} \times \mathbf{b}_0 \quad &\text{on } \partial \Omega, \\
r &= r_0 \quad &\text{on } \partial \Omega,
\end{align*}
\]

with \( \mathbf{n} \) being the unit outward normal on \( \partial \Omega \) and \( \mathbf{u}_0, \mathbf{b}_0, \) and \( r_0 \) being the functions defined on the boundary.

Upon finite element discretization, using appropriate spaces and Picard linearization, we obtain the following linear system:

\[
\begin{pmatrix}
F & B^T & C^T & 0 \\
B & 0 & 0 & 0 \\
-C & 0 & M & D^T \\
0 & 0 & D & 0
\end{pmatrix}
\begin{pmatrix}
\delta u \\
\delta p \\
\delta b \\
\delta r
\end{pmatrix}
= \begin{pmatrix}
r_u \\
r_p \\
r_b \\
r_r
\end{pmatrix},
\]

with

\[
\begin{align*}
r_u &= f - Fu_k - C^T b_k - B^T p^k, \\
r_p &= -Bu_k, \\
r_b &= g - Mu_k + Cb_k - D^T r^k, \\
r_r &= -Db_k,
\end{align*}
\]

where \( F \) is a discrete convection-diffusion operator, \( B \) is a fluid divergence operator, \( M \) is the curl-curl operator, \( D \) is the magnetic divergence operator, and \( C \) represents the coupling terms.
The linear system needs to be solved repeatedly with changing right-hand-sides throughout the nonlinear iteration. It is possible to formulate the Newton iteration with a minor additional effort. In recent years, interest in the development of block preconditioning methods for the MHD model has increased; see, for example, [1, 4, 5, 6, 7]. In this talk we introduce a new approximate inverse preconditioner. The derivation exploits the nullity of the discrete curl-curl operator in the Maxwell subproblem. Using a result proven in [2], we show that the inverse of the coefficient matrix contains zero blocks, and use discretization considerations to obtain a practical preconditioner based on further sparsification.

We perform a careful analysis that shows that a few of the eigenvalues of the preconditioned matrix have a high algebraic multiplicity. A notable property of the preconditioner is that it is inherently indefinite, and while the problem is nonsymmetric and the eigenvalues of the preconditioned matrix are complex, it can be shown that their real parts are all positive and a large number of them tend to cluster around the value 1.

We demonstrate the scalability of the new preconditioner with a set of numerical experiments, including large three-dimensional problems. Full results are available in [8].

References


A multilevel Schwarz preconditioner based on a hierarchy of robust coarse spaces


Inria Paris, Laboratoire J. L. Lions, Sorbonne Université
https://who.rocq.inria.fr/Laura.Grigori/

Abstract

We consider the solution of a linear system of equations $Ax = b$, where $A \in \mathbb{R}^{n \times n}$ is a symmetric positive definite (SPD) matrix, $b \in \mathbb{R}^n$ is the right-hand side, and $x \in \mathbb{R}^n$ is the vector of unknowns. To enhance convergence, it is common to solve the preconditioned system

$$M^{-1}Ax = M^{-1}b.$$ 

Standard domain decomposition preconditioners such as block Jacobi, additive Schwarz, and restricted additive Schwarz methods are widely used and they are highly parallel. However, their role in lowering the condition number of $M^{-1}A$ typically deteriorates when the number of subdomains increases. Robust two-level Schwarz preconditioners based on coarse space corrections exist in the literature and can provide robust convergence estimates, see e.g. [1]. There is a rich literature on this topic, due to space limits we are not able to cite it here, but for more details please refer to [3]. As long as the dimension of the coarse space is reasonable, that is, exact solvers can be used efficiently, two-level methods scale well on parallel architectures. But the factorization of the coarse space matrix may become costly at scale. An alternative is then to use an iterative method on the second level, combined with an algebraic preconditioner, such as a one-level additive Schwarz preconditioner. Nevertheless, the condition number of the resulting preconditioned coarse space matrix may still be large. One of the difficulties of using more advanced methods, like algebraic multigrid or even two-level overlapping Schwarz methods, to solve the coarse problem is that the matrix does not arise from a partial differential equation (PDE) anymore.

We present a novel robust multilevel additive Schwarz preconditioner where at each level the condition number is bounded, ensuring a fast convergence for each nested solver. It is based on a hierarchy of robust coarse spaces that are able to transfer spectral information from one level to the next. In our work, the only information from the PDE needed for the construction of the preconditioner consists of the local Neumann matrices at the fine level, as in Geneo preconditioner [1]. These matrices correspond to the integration of the bilinear form in the weak formulation of the studied PDE on the subdomain-decomposed input mesh. No further information is necessary: except on the fine level, our method is algebraic and does not depend on any coarsened mesh or auxiliary discretized operator. This preconditioner is presented in more details in [3] on which this abstract is based, the code to reproduce the results presented in [3] is available at https://github.com/prj-/aldaas2019multi. The parallel multilevel preconditioner is implemented in HPDDM solver and is available now in Petsc through the PCHPDDM routine, for more details see the associated website [4].

Our preconditioner is defined as following. At the first level, the set of unknowns is partitioned into $N_1$ subdomains and each subdomain has an associated matrix $A_{1,j} = R_{1,j}AR_{1,j}^\top$ obtained by using appropriate restriction and prolongation operators $R_{1,j}$ and $R_{1,j}^\top$ respectively. The preconditioner is formed as an additive Schwarz preconditioner coupled with an additive coarse space correction,
defined as,

\[ M^{-1} = M_1^{-1} = V_1 A_2^{-1} V_1^T + \sum_{j=1}^{N_1} R_{1,j}^T A_{1,j}^{-1} R_{1,j}, \]

where \( V_1 \) is a tall-and-skinny matrix spanning a coarse space obtained by solving for each subdomain \( j = 1 \) to \( N_1 \) a generalized eigenvalue problem involving the matrix \( A_{1,j} \) and the Neumann matrix associated with subdomain \( j \). The coarse space matrix is \( A_2 = V_1^T A V_1 \). This is equivalent to the GenEO preconditioner, and is described in detail in [1]. The dimension of the coarse space is proportional to the number of subdomains \( N_1 \). When it increases, factorizing \( A_2 \) by using a direct method becomes prohibitive, and hence the application of \( A_2^{-1} \) to a vector should also be performed through an iterative method.

Our multilevel approach defines a hierarchy of coarse spaces \( V_i \) and coarse space matrices \( A_i \) for \( i = 2 \) to any depth \( L + 1 \), and defines a preconditioner \( M_i^{-1} \) such that the condition number of \( M_i^{-1} A_i \) is bounded. The depth \( L + 1 \) is chosen such that the coarse space matrix \( A_{L+1} \) can be factorized efficiently by using a direct method. At each level \( i \), the graph of the coarse space matrix \( A_i \) is partitioned into \( N_i \) subdomains, and each subdomain \( j \) is associated with a local matrix \( A_{i,j} = R_{i,j} A_{i} R_{i,j}^T \) obtained by using appropriate restriction and prolongation operators \( R_{i,j} \) and \( R_{i,j}^T \), respectively. The preconditioner at level \( i \) is defined as,

\[ M_i^{-1} = V_i A_{i+1}^{-1} V_i^T + \sum_{j=1}^{N_i} R_{i,j}^T A_{i,j}^{-1} R_{i,j}, \]

where the coarse space matrix is \( A_{i+1} = V_i^T A_i V_i \).

One of the main contributions of the paper concerns the construction of the hierarchy of coarse spaces \( V_i \) for levels \( i \) going from 2 to \( L \), that are built algebraically from the coarse space of the previous level \( V_{i-1} \). This construction is based on the definition of local symmetric positive semi-definite (SPSD) matrices associated with each subdomain \( j \) at each level \( i \) that we introduce in [2] and extend in [3]. These matrices are obtained by using the local SPSD matrices of the previous level \( i - 1 \) and the previous coarse space \( V_{i-1} \). They are then involved, with the local matrices \( A_{i,j} \), in concurrent generalized eigenvalue problems solved for each subdomain \( j \) that allows to compute the local eigenvectors contributing to the coarse space \( V_i \). We show in [3], that the condition number of \( M_i^{-1} A_i \) is bounded and depends on the maximum number of subdomains at the first level that share an unknown, the number of distinct colors required to color the graph of \( A_i \) so that \( \{ \text{span}(R_{i,j}) \}_{1 \leq j \leq N_i} \) of the same color are mutually \( A_i \)-orthogonal, and a user defined tolerance \( \tau \). It is thus independent of the number of subdomains \( N_i \). We illustrate the theoretical robustness and practical usage of our proposed method by performing strong scalability tests on up to 8192 processes for a three-dimensional linear elasticity problem of size \( 616 \times 10^6 \).

**References**


Abstract

This talk is motivated by numerical problems in modeling and simulation of elastic stents, [4]. To this end, we will present a general approach to the dynamic (hyperbolic problem) and stationary simulation of elastic frame structures. As prototypes we will present two 1D models of an endovascular stent. Endovascular stents are biomedical devices made of struts and are used for treating arterial stenosis. The state of the system in both models is described by a vector valued function on a metric graph which satisfies a system of ODEs and a set of algebraic constraints. Both models are obtained, and thus theoretically validated, by Γ-convergence from 3D nonlinear elasticity. As a result of asymptotic analysis, solutions are contained in a space of functions which are constrained by a set of algebraic conditions in the nodes of the graph and by requiring that the middle line of a strut does not extend. In the first model, see [1], we place all constraints in the variational product space and build a finite element approximation there, whereas in the second model, see [2], we study the problem in a large free product space and leave all of the constraints as additional variables (e.g. continuity of displacements and of micro rotations) as a part of the system matrix to be removed by a linear algebra solver in a search of the solution. We will present convergence results for both models, but the much more puzzling question is which of the models will yield more efficient numerical methods? Namely, the second model yields a system matrix which is more than three times larger than in the first model. As a benchmark solution method we will use the block LDL* decomposition as implemented in MA57 from [3]. It turns out that the running times on a smaller matrix can be as much as 23 times longer than on the extended matrix (whose dimension is three times as large as for the small matrix). We will analyze this observation by relating the pivoting strategy to the structure of a system matrix as influenced by the modeling approach. The extended matrix structurally reflects much more directly the adjacency pattern of the topological graphs of a stent.

In the second part of the talk we will concentrate on the dynamic problem. An abstract dynamic problem reads: given spaces $V_S$ and $Q_S$ find a $V_S$ valued function $u_S$ so that

$$
\frac{d^2}{dt^2} m_S(u_S, \ddot{u}_S) + k_S(u_S, \dot{u}_S) + b_S(n_S, \dot{u}_S) = l_S(\dot{u}_S), \quad \dot{u}_S \in V_S,
$$

$$
b_S(\ddot{n}_S, u_S) = 0, \quad \ddot{n}_S \in Q_S.
$$

An important feature of (1) is that the form $m_S$ is only positive semidefinite, and so we have a differential algebraic system of equations, [8, 7]. We will use the results of the numerical analysis of the stationary problem coupled with the canonical form construction for a DAE as a starting point for the analysis of the resolvent of the associated operator pencil. If time permits we will also consider an abstract infinite dimensional setting based on the aforementioned algebraic manipulation and the analysis of the representation theorems for indefinite forms and associated operators from [5]. Also, we will discuss the spectral structure of the associated operator pencil.

Let us point out a further advantage of the extended model. Given that the constraints do not have to be incorporated in the function spaces, as is typically the case, the analysis of the inf-sup
constant for this system of partial differential equations greatly simplifies, in particular a proof of an inf-sup continuous inequality directly transfers to a discrete inf-sup inequality in the discretized setting. Let us note that we did not have a discrete inf-sup in the basic model from [1, 4]. Based on these results we retrace the steps of the error analysis from [6] and present the corresponding results. We also discuss the numerical stability and the efficiency optimization of the time-stepper for (1) based on a recycling strategy of the $LDL^*$ decomposition.

Finally, we present validation experiments for the method by comparing it empirically to the 3D model solved by the standard legacy finite element code.

References


Computing the approximate common divisor of polynomials

Antonio Fazzi, Nicola Guglielmi, and Ivan Markovsky

Section of Mathematics, Gran Sasso Science Institute
https://www.gssi.it/people/professors/lectures-maths/item/545-guglielmi-nicola

Abstract

Computing the greatest common divisor of a set of polynomials is a problem appearing in different fields, such as linear control and network theory. In practice, the polynomials are obtained through measurements, so that their coefficients are inexact. This poses the problem of computing an approximate greatest common divisor (GCD). We propose a method which restates the problem as a (structured) distance to singularity of an associated Sylvester matrix. The algorithm naturally applies to a finite set polynomials and extends to compute an approximate GCD of degree larger than 1. If the polynomials have real coefficients then the method assures that the computed nearby polynomials having a common GCD have real coefficients.

Let us consider for simplicity the case of two polynomials, \( p \in \mathcal{P}_n \) and \( q \in \mathcal{P}_m \) (with \( \mathcal{P}_k \) the set of polynomials of degree at most \( k \)),

\[
\begin{align*}
p(z) &= a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0 \\
q(z) &= b_m z^m + b_{m-1} z^{m-1} + \cdots + b_1 z + b_0
\end{align*}
\]

and similarly \( \hat{p} \) and \( \hat{q} \) of the same degrees, respectively. Denote by

\[
\begin{align*}
a &= (a_n \ a_{n-1} \ \cdots \ a_1 \ a_0)^T \\
b &= (b_m \ b_{m-1} \ \cdots \ b_1 \ b_0)^T
\end{align*}
\]

the vectors of the coefficients of the polynomials \( p \) and \( q \), respectively, and similarly \( \hat{a} \) and \( \hat{b} \) be the vectors of the coefficients of the polynomials \( \hat{p} \) and \( \hat{q} \), respectively. Then, we define the distance measure

\[
\text{dist} \left( (p, q), (\hat{p}, \hat{q}) \right) = \sqrt{\sum_{i=0}^{n} |a_i - \hat{a}_i|^2 + \sum_{j=0}^{m} |b_j - \hat{b}_j|^2}
\]

that is the Euclidean norm of the vector \( \left( \frac{a - \hat{a}}{b - \hat{b}} \right) \).

The problem we consider is defined as follows.

**Problem 1** Given a pair of coprime polynomials \( p \in \mathcal{P}_n, q \in \mathcal{P}_m \) find:

\[
\hat{C}(p, q) = \inf_{\hat{p} \in \mathcal{P}_n, \hat{q} \in \mathcal{P}_m} \text{dist} \left( (p, q), (\hat{p}, \hat{q}) \right) \quad (\text{coprimeness radius}).
\]

It is easy to see that Problem 1 can be naturally reformulated as the structured distance to singu-
larity of the Sylvester matrix

\[
S = \begin{pmatrix}
a_n & \ldots & a_m & \ldots & \ldots & a_1 & a_0 & 0 & \ldots & 0 \\
0 & a_n & \ldots & a_m & \ldots & \ldots & a_1 & a_0 & 0 & \ldots \\
\vdots & 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
0 & \ldots & 0 & a_n & \ldots & \ldots & a_1 & a_0 & 0 & \ldots \\
0 & \ldots & 0 & b_m & \ldots & b_1 & b_0 & 0 & \ldots & 0 \\
0 & \ldots & 0 & b_m & \ldots & b_1 & b_0 & 0 & \ldots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
0 & \ldots & 0 & 0 & b_m & \ldots & b_1 & b_0 & 0 & \ldots \\
0 & \ldots & 0 & 0 & 0 & b_m & \ldots & b_1 & b_0 & 0 \\
\end{pmatrix}
\]

in the Frobenius norm.

The topic of computing a so-called \( \varepsilon \)-GCD of a pair of polynomials \( p \in \mathcal{P}_n \) and \( q \in \mathcal{P}_m \), that is a nearby pair of polynomials \( \hat{p} \in \mathcal{P}_n, \hat{q} \in \mathcal{P}_m \) having a non-trivial GCD, has been studied in the literature, where several criteria have been used in order to specify the nearness property. In many cases, for given polynomials \( p \) and \( q \), and a tolerance \( \varepsilon \), the methods aim to find the degree of an \( \varepsilon \)-GCD, a set of perturbations \( \delta p, \delta q \) (such that \( \hat{p} = p + \delta p, \hat{q} = q + \delta q \)) and an \( \varepsilon \)-GCD (w.r.t. the perturbations) without addressing the minimization in (3) directly. Here instead we look for the pair with minimal distance (in the 2-norm) and allow further constraints on the coefficients of the polynomials.

Problem 1 is a non-convex optimization problem and can be approached by global optimization, local optimization, and convex relaxation methods. The methods based on global optimization, such as the branch and bound method, are too expensive for most real-life problems. Here we consider the local optimization approach. Our main contribution [1, 2] is a new method based on the integration of a constrained gradient system of ordinary differential equations, which describes the gradient dynamics associated to an appropriate cost functional, which is given by the modulus of the smallest eigenvalue (or by the smallest singular value) of the associated Sylvester matrix.

The method is globally convergent to a locally optimal solution. Our simulation results indicate that it is more robust to the initial approximation than the Newton-type methods. In addition, we incorporate the additional constraint of the exact knowledge of some coefficients of the polynomials \( p \) and \( q \) into the method and are able to treat some extensions of Problem 1.

A generalization to matrix polynomials has also been recently considered [3].

References


Sharp 2-Norm Error Bounds for the Conjugate Gradient Method and LSQR

Eric Hallman

Department of Mathematics, North Carolina State University
https://erhallma.math.ncsu.edu

Abstract

We consider using the conjugate gradient method (CG) to solve the problem

$$Ax = b,$$

(CG)

where $A$ is an $n \times n$ symmetric positive definite matrix. Assuming $x_0 = 0$, CG produces an iterate at the $k$th step that minimizes the energy norm error $\|x_k - x_*\|_A$ over the Krylov subspace

$$K_k(A, b) = \text{Span}\{b, Ab, \ldots, A^{k-1}b\},$$

where $x_* = A^{-1}b$. In exact arithmetic, the method will terminate in at most $n$ steps. This often fails to happen in practice due to roundoff errors, so our goal is to monitor the 2-norm error $\|x_k - x_*\|_2$ in order to determine when the algorithm may be safely halted. As was the case for previous work in bounding the 2-norm error for CG, we assume exact arithmetic and require a nontrivial lower bound $\tilde{\lambda}$ on the smallest eigenvalue of $A$.

The 2-norm error for CG is known to decrease monotonically, although it is not minimized over $K(A, b)$. Furthermore, the iterates $x_k$ are known to update along positively correlated directions, so that if $x_k = x_{k-1} + p_k$ then $p_i^T p_j > 0$ for all $i$ and $j$. Recently, Estrin, Orban, and Saunders [1] developed an error bound that took advantage of these properties. They established a bound of the form

$$\|x_*\|_2 \leq \|\tilde{z}_k\|_2$$

(1)

where $\tilde{z}_k \in \mathbb{R}^k$ is cheap to compute and depends on the eigenvalue bound $\tilde{\lambda}$. They then derived an upper bound on the CG error by noting that

$$\|x_k - x_*\|_2 \leq \left(\|\tilde{z}_k\|_2^2 - \|x_k\|_2^2\right)^{1/2},$$

and showed how to use the related algorithm SYMMLQ as an auxiliary to CG in order to compute this quantity in a more stable manner.

Other work by various authors including Golub, Meurant, Tichý, and Strakoš [2, 4, 3, 5] has worked to compute an upper bound for the $A$-norm of the CG error. This bound may be used to derive the 2-norm error bound

$$\|x_k - x_*\|_2 \leq \tilde{\lambda}^{-\frac{1}{2}}\|x_k - x_*\|_2 < \tilde{\phi}_{k+1} \tilde{\lambda}^{-\frac{1}{2}},$$

(2)

where $\tilde{\phi}_{k+1}$ is a cheaply computable quantity depending on $\tilde{\lambda}$.

Our work connects and improves upon these two approaches. We identify the iterate $\tilde{z}_{k+1}$ with a point $\tilde{x}_{k+1} \in K_{k+1}(A, b)$, so that (1) may be extended by one step and rewritten in the form

$$\|x_*\|_2 \leq \|\tilde{x}_{k+1}\|_2.$$

We tighten this bound by showing that the solution $x_*$ lies within an ellipsoid centered at the point

$$\tilde{x}_{k+1}^E = \frac{1}{2}(x_k + \tilde{x}_{k+1}).$$
One axis of this ellipsoid is the line segment between \( x_k \) and \( \tilde{x}_{k+1} \), and the point that minimizes the 2-norm error over \( K_{k+1}(A,b) \) falls on this line segment.

We use this result to derive new error bounds for CG. We also note that the error bounds for \( \tilde{x}_{k+1}^{E} \) are even tighter than the corresponding CG bounds (although \( \| \tilde{x}_{k+1}^{E} - x_* \|_2 \) is not necessarily smaller than \( \| x_k - x_* \|_2 \)), and in particular find that

\[
\| \tilde{x}_{k+1}^{E} - x_* \|_2 \leq \frac{1}{2} \left| \tilde{\phi}_{k+1} \right| \lambda^{-\frac{1}{2}},
\]

which is precisely a factor of 2 smaller than the bound in (2).

Although our results offer only a minor improvement over the existing ones, we are able to show that they are, in a sense, sharp. Our bound relies only on \( \tilde{\lambda} \) and the information gained by running \( k \) steps of CG, and no upper bound on the 2-norm error can do better without using more information. Thus future work aimed at developing stopping criteria must either use more information about the structure of the problem to be solved, or rely on error estimates that are not guaranteed to be upper bounds.

We derive similar error bounds when using LSQR or Craig’s method to solve

\[
\min_x \| Ax - b \|_2
\]

or

\[
\min_x \| x \|_2 \text{ subject to } Ax = b,
\]

since in exact arithmetic LSQR is equivalent to CG run on the normal equations. As was the case for CG, the existing error bounds are nearly as tight as can be expected using current approaches. We present numerical experiments on examples from the SuiteSparse matrix collection, and discuss some practical limitations of our methods.

This abstract is based on work that has been submitted for review at the SIAM Journal of Matrix Analysis and Applications.

References


Algorithms with automatic result verification for the matrix exponential

Andreas Frommer, and Behnam Hashemi

Department of Mathematics, Shiraz University of Technology
https://sites.google.com/view/bhashemi

Abstract

One of the most important matrix functions is the matrix exponential. Its applications include the solution of different types of differential equations, Markov models and control theory; see e.g., [2] and references therein. Most of the time in practice, one is only able to compute an approximation to the exact matrix exponential. This is due to truncation errors, as well as rounding errors which are intrinsic to algorithms in floating point arithmetic. Thus, a great deal of attention is devoted to derive error bounds (which are typically a priori and normwise), and to backward error analysis of algorithms for the matrix exponential. Backward error bounds, when combined with condition numbers, could result in estimates for the forward error.

A very different approach to forward error analysis is the use of algorithms with automatic result verification. Such algorithms rely on machine interval arithmetic to compute mathematically rigorous forward error bounds. We provide a review of old and present new algorithms with automatic result verification to compute mathematically correct error bounds for each entry of the matrix exponential. These algorithms can be used as an automatic a posteriori forward error analysis tool to monitor the accuracy of floating point approximations, e.g., those obtained using MATLAB’s expm. In what follows we outline the main idea behind a Padé approximation method, a contour integration approach, and a Chebyshev approximation method which is designed for Hermitian matrices. All of these three algorithms are new in the context of verification methods.

Consider the \((k,m)\)-type Padé method

\[
\exp(A) = q_m(A)^{-1} p_k(A) + r_{km}(A),
\]

where \(p_k\) and \(q_m\) are polynomials of degree \(k\) and \(m\), respectively, and \(r_{km}\) is the corresponding reminder term which in the scalar case satisfies \(r_{km}(x) = \mathcal{O}(x^{k+m+1})\). Obviously,

\[
q_m(A) \exp(A) = p_k(A) + T_{km},
\]

where \(T_{km} := q_m(A) r_{km}(A)\). Our Padé-type verification method relies on the following theorem to bound every entry of the matrix \(T_{km}\).

**Theorem 1** [1] Let \(\|\cdot\|\) be any submultiplicative matrix norm. Then

\[
\|T_{km}\| \leq \pi(k, m, \|A\|) := \frac{k! \, m!}{(k + m)! \, (k + m + 1)!} \|A\|^{k+m+1} \exp(\|A\|).
\]

In practice, we compute interval matrices \(P \ni p_k(A)\) and \(Q \ni q_m(A)\) and a rigorous upper bound \(\pi\) for \(\pi(k, m, \|A\|)\) obtained using interval arithmetic. We then compute an interval matrix that contains the solution set of the interval linear system

\[
Q \cdot X = P + \pi E,
\]

where \(E\) denotes the interval matrix with all entries equal to \([-1, 1]\). Therefore, the computed solution set contains the exact matrix exponential \(\exp(A)\).

On the other hand, our verification method based on contour integration relies on the following theorem.
Theorem 2 [1] Let \( \|A\| < 1 \) where \( \|\cdot\| \) is the 1-, 2- or \( \infty \)-norm, and let \( c \) be such that \( e^{-c} > 2\|A\| \). Let \( z_k = e^{2\pi ik/N} \), \( k = 1, \ldots, N \) where \( N \in \mathbb{N} \). Then, with
\[
\gamma(c, N) := \frac{4e^{2c}\exp(e^c)}{e^{cN} - 1},
\]
we have
\[
\exp(A) \in \frac{1}{N} \sum_{k=1}^{N} z_k \exp(z_k)(z_kI - A)^{-1} + \gamma(c, N)E.
\]
In practice, we use INTLAB’s \texttt{verifylss} [3] to compute an enclosure for each of the exact inverses \( (z_kI - A)^{-1} \), and a rigorous upper bound \( \gamma \) for \( \gamma(N, c) \). Here \( c \) is chosen such that \( N \) is minimal under all pairs \((N, c)\) which satisfy
\[
\gamma(c, N) \leq 10 \varepsilon_{\text{mach}},
\]
where \( \varepsilon_{\text{mach}} \) is the machine epsilon.
Finally, let \( I_k(t) \) and \( T_k(x) \) denote the modified Bessel functions of the first kind and Chebyshev polynomials of the first kind, respectively. Our verified Chebyshev approximation method is based on the following result.

Theorem 3 [1] Let \( A \) be Hermitian with spectrum in \([-1, 1]\) and let \( p_d(A) \) be the degree \( d \) truncated Chebyshev series approximation
\[
p_d(A) = I_0(1)I + \sum_{k=1}^{d} 2I_k(1) \cdot T_k(A)
\]
for \( \exp(A) \). Then, with \( \tau(\rho, d) \) defined for \( \rho \geq 1 \) as
\[
\tau(\rho, d) := 2e^{\rho \frac{\rho - 1}{\rho - d}}
\]
we have
\[
\exp(A) \in p_d(A) + \tau(\rho, d)E.
\]
We give a quick review of the basics of machine interval arithmetic. Then, we present implementation details including the choice of parameters in each method, and a scheme for efficient squaring of interval matrices which is important for our algorithms as they rely on the scaling and squaring framework. A variety of numerical experiments and comparisons concludes this talk.

References

Cluster Gauss-Newton method for finding Multiple Global Minimizers of Nonlinear Least Squares problems - with application to Parameter Estimation of Pharmacokinetic Models

Yasunori Aoki, Ken Hayami, Kota Toshimoto, and Yuichi Sugiyama

National Institute of Informatics, and The Graduate University for Advanced Studies, SOKENDAI
https://researchmap.jp/KenHayami/

Abstract

1 Introduction

Parameter estimation problems of mathematical models can often be formulated as nonlinear least squares problems. Typically these problems are solved numerically using iterative methods. The local minimizer obtained using these methods usually depends on the choice of the initial iterate. The estimated parameter and subsequent analyses using the estimated parameter depends on the choice of the initial iterate. One way to reduce the analysis bias due to the choice of the initial iterate is to repeat the algorithm from multiple initial iterates (i.e., use a multi-start method). However, the procedure can be computationally intensive and does not always give global minimizers.

To overcome this problem, we propose the Cluster Gauss-Newton (CGN) method[1, 2], an efficient algorithm for finding multiple global minimizers of nonlinear least squares problems. CGN simultaneously solves the nonlinear least squares problem from multiple initial iterates. Then, CGN iteratively improves the solutions from these initial iterates similarly to the Gauss-Newton method. However, it uses a global linear approximation instead of the Jacobian. The global linear approximations are computed collectively among all the current iterates in order to seek global minimizers, and to minimize computational costs.

We use mathematical models used in pharmaceutical drug development to demonstrate its use and to show that the proposed algorithm is computationally more efficient compared to the standard Levenberg-Marquardt method, state of the art multi-start methods, and derivative-free methods.

2 Nonlinear least squares problem of our interest

In this talk, we propose an algorithm for obtaining multiple global minimizers of nonlinear least squares problems.

\[
\min_{x} \|f(x) - y^*\|_2^2
\]

which do not have a unique solution (global minimizer). Such a problem arises in the parameter estimation of a mathematical model of pharmaceutical drug concentration in a human body called the physiologically based pharmacokinetic (PBPK) model. The PBPK model is a system of mildly nonlinear stiff ordinary differential equations (ODEs) with many parameters.

Here, \(x\) is the vector consisting of the unknown parameters of the ODEs, and \(f(x)\) is obtained by solving the ODEs in time to obtain e.g. the drug concentrations in the blood plasma at certain time intervals, which should correspond to the vector of observed values \(y^*\).
The model is constructed based on the knowledge of the mechanism of how the drug is absorbed, distributed, metabolised and excreted. Given the complexity of this process and the limitation of the observations one can obtain, the model parameters cannot be uniquely identified from the observations, so that there are non-unique global minimizers to the least squares problem.

3 The CGN method

The method is described as follows.

1) Create initial cluster of solutions

Set $k := 0$. Generate the initial solution vectors $\{x_i^{(0)}\}_{i=1}^{N}$ randomly following a uniform distribution within a plausible domain given by $x^L \leq x \leq x^U$.

Evaluate the nonlinear function $f$ at each $x_i^{(0)}$ as $y_i^{(0)} = f(x_i^{(0)})$, $i = 1, \cdots, N$.

2) Main iteration

Construct weighted linear approximations of the nonlinear function around each point $x_i^{(k)}$, $i = 1, \cdots, N$, such that,

$$f(x) \approx A^{(k)}_{(i)}(x - x_i^{(k)}) + f(x_i^{(k)}),$$

where $A^{(k)}_{(i)} \in \mathbb{R}^{m \times n}$ is given by

$$A^{(k)}_{(i)} = \arg\min_{A \in \mathbb{R}^{m \times n}} \sum_{j=1}^{N} \left\| d^{(k)}_{j(i)} \left\| f(x_j^{(k)}) - \left\{ A \left( x_j^{(k)} - x_i^{(k)} \right) + f(x_i^{(k)}) \right\} \right\|_2^2 \right(3)$$

The key difference between CGN and other methods is that we construct the matrix $A^{(k)}_{(i)}$ collectively using all the function values $f(x_j^{(k)})$, $j = 1, \ldots, N$, at the iterates of the previous iteration, instead of computing the Jacobian at each iterate. The weights $d^{(k)}_{j(i)}$ in (3) are chosen as

$$d^{(k)}_{j(i)} = \begin{cases} \left( \frac{1}{\sum_{l=1}^{N}(x_{ij}^{(k)} - x_{il}^{(k)})/(x_j^U - x_j^L))^2} \right)^\gamma & \text{if } j \neq i, \\ 0 & \text{if } j = i. \end{cases} \tag{4}$$

where $\gamma \geq 0$ is a constant, e.g. $\gamma = 1$.

3) Update $x_i$:

$$x_i^{(k+1)} = x_i^{(k)} + \left( A^{(k)}_{(i)}^T A^{(k)}_{(i)} + \lambda_i^{(k)} I \right)^{-1} A^{(k)}_{(i)}^T (y^* - y_i^{(k)}), \quad i = 1, \cdots, N,$$

Here, $\lambda_i^{(k)}$ are the regularization parameters for controlling the step lengths.

4) Update $y_i$:

$$y_i^{(k+1)} = f(x_i^{(k+1)}), \quad i = 1, \cdots, N$$

5) Set $k := k + 1$ and iterate until convergence.

References


GUI software: http://www.bluetree.me/CGNmethod_for_PBPKmodels

Matlab code available in Matlab central file exchange
Extracting relevant data from linear data fitting problems via generalized Krylov subspace methods

Iveta Hnětynková, Martin Plešinger, and Jana Žaková

Faculty of Mathematics and Physics, Charles University
http://www.ms.mff.cuni.cz/~hnetynka/

Abstract
In various areas, there is a need to solve linear data-fitting problems

\[ AX \approx B, \quad A \in \mathbb{R}^{m \times n}, \ B \in \mathbb{R}^{m \times d}, \]

where \( A \) is the given model and \( B \) is the multiple-observation (measurement) matrix. Depending on the particular application, the problem above typically suffers from the presence of various types of errors, and also irrelevant or redundant data in \( B \) and/or \( A \) making the system incompatible. Least squares methods (such as ordinary, total or scaled least squares) are the most widely used here, where generally a minimum norm data correction is searched ensuring existence of the approximate solution \( X \). However, such a correction may not always exist, as described for example for total least squares with \( d = 1 \) already in the early work [2]. Additional difficulties for problems with \( d > 1 \) are related to the fact that the individual observations (columns of \( B \)) can be closely correlated with different subsets of columns of \( A \), etc.

Core reduction
The core reduction introduced in [7] and [8] for single-observation case, \( d = 1 \), represents a fundamentally different concept. It suggests to extract the necessary and sufficient information for solving the data-fitting problem into the so called core problem, while moving the redundant and/or irrelevant information out. The core reduction was extended to multiple-observation problems in [4]. It was proved that for any data \([B|A]\) there exist orthogonal matrices \( P \in \mathbb{R}^{m \times m}, Q \in \mathbb{R}^{n \times n}, R \in \mathbb{R}^{d \times d} \) such that

\[
P^T[B|A]\begin{bmatrix} R & 0 \\ 0 & Q \end{bmatrix} = P^T[BR|AQ]\begin{bmatrix} B_1 & 0 \\ 0 & A_{11} \\ 0 & 0 \\ 0 & A_{22} \end{bmatrix},
\]

where \( A_{11} \) and \( B_1 \) are minimally dimensioned subject to the orthogonal transformation above. Assuming orthogonal invariance of the considered least squares minimization, only the core problem \( A_{11}X_1 \approx B_1 \) has to be solved, complemented by the back transformation \( X \equiv Q \begin{bmatrix} X_1 & 0 \\ 0 & 0 \end{bmatrix} R^T \), see [4]. Recently, our papers [5] and [6] introduced further generalizations of the core reduction to data-fitting problems, where the observation matrix \( B \) or the matrix \( A \) of the model has tensor (or some other special) structure.

Iterative methods based on Golub-Kahan bidiagonalization
The fundamental question is the determination of core problems for data, where the singular value decomposition or Tucker decomposition (used for the derivation of the core reduction) are not suitable. Assuming exact arithmetic, it was shown in [8] that for \( d = 1 \), the Golub-Kahan iterative bidiagonalization [1] can be used. Let \( w_0 = 0 \) and \( s_1 = B/\beta_1 \), where \( \beta_1 = \|B\| \). The algorithm
computes for \( k = 1, 2, \ldots \)

\[
\alpha_k w_k = A^T s_k - \beta_k w_{k-1}, \quad \|w_k\| = 1,
\]

\[
\beta_{k+1} s_{k+1} = Aw_k - \alpha_k s_k, \quad \|s_{k+1}\| = 1,
\]

until one of the normalization coefficients \( \alpha_k \geq 0, \beta_{k+1} \geq 0 \) becomes zero. The multiple-observation case, \( d > 1 \), was studied mathematically in [4]. It was shown that a particular band generalization of the algorithm above yields after \( d \) deflations the core problem in the form of a band matrix. In the incompatible case of our interest, we obtain for \( d = 1 \) and \( d > 1 \), e.g.,

\[
[B_1 | A_{11}] = \begin{bmatrix}
\beta_1 & \alpha_1 & & \\
\beta_2 & \ddots & \alpha_{k-1} & \\
& \ddots & \ddots & \beta_k
\end{bmatrix}
\quad \text{and} \quad
\begin{bmatrix}
\beta_1 & \delta_{1,2} & \delta_{1,3} & & \\
\delta_{2,1} & \beta_2 & \alpha_2 & \alpha_3 & \\
\delta_{3,1} & \beta_3 & \delta_{3,2} & \alpha_4 & \\
& \delta_{4,2} & \delta_{4,3} & \beta_4 & \alpha_5 & \beta_5 & \delta_{5,4} & \alpha_6 & \delta_{6,5} & \beta_7 & \alpha_7
\end{bmatrix},
\]

respectively.

**Finite precision computation and further generalizations**

In practical computations, however, the use of short recurrences in the Golub-Kahan bidiagonalization and its generalizations leads to well known effects such as the loss of global orthogonality, loss of linear independence among the computed vectors etc. Consequently, the normalization coefficients may not become close to zero, exact deflation is not possible and revealing of the core problem complicates significantly. In this presentation we adress these issues. In particular, we concentrate on the question how well can the core problem be approximated by the considered algorithms.

Furthermore, we present iterative approaches arising from further generalizations of the Golub-Kahan bidiagonalization, for determination of the core problem for selected generalized data-fitting problems introduced in [5] and [6]. These algorithms are currently under investigation.

**References**


NEP-PACK: A Julia package for nonlinear eigenproblems

Elias Jarlebring, Max Bennedich, Giampaolo Mele, Emil Ringh, Parikshit Upadhyaya

KTH Royal Institute of Technology
www.math.kth.se/~eliasj

Abstract

We present NEP-PACK, a novel open-source library for the solution of nonlinear eigenvalue problems (NEPs), defined as the problem of determining the singular points of a matrix: find \((\lambda, v) \in \mathbb{C} \times \mathbb{C}^n \setminus \{0\}\) such that

\[ M(\lambda)v = 0 \]

where \(M : \mathbb{C} \to \mathbb{C}^{n \times n}\) is a meromorphic function. The package is implemented in the Julia language. It is available in the Julia package ecosystem (under the name NonlinearEigenproblems.jl).

We have developed data-types to represent NEPs, techniques to manipulate the data types, as well as efficient implementations of many state-of-the-art algorithms. The package makes full use of the efficiency of Julia, yet maintains usability, and integrates well with other software packages and other programming languages (including MATLAB). The package is designed to be easy to use for application researchers as well as algorithm developers. The data-structures are chosen with particular attention to algorithm neutrality, in order to simplify the study and compare algorithm properties. The software development is done in an open source manner, using modern software engineering tools (such as continuous integration with unit testing). Tutorials, software and examples are available online: https://nep-pack.github.io/NonlinearEigenproblems.jl/

Algorithms and gallery problems. Currently, the package contains implementations of the following NEP-algorithms: Jacobi-Davidson (two versions: jd_ffenberger, jd_betcke), Nonlinear Arnoldi method, Beyn’s contour integral method, Block SS contour integration, NLEIGS\(^1\), Infinite Arnoldi method, Tensor infinite Arnoldi method, Infinite bi-Lanczos, Infinite Lanczos, Classical Newton-Raphson, Augmented Newton, Residual inverse iteration, Quasi-Newton, Block Newton, Rayleigh functional iteration (version a, b), Newton-QR, Implicit determinant method and Broyden’s method.

Common features and subtasks of different algorithms are implemented by code sharing, e.g., the contour integral methods have a common way to compute the integrals, such that if a new quadrature method is implemented, it can automatically be used for both contour integral methods. The separation of subtask, allow for user and application specific improvements including parallelization, e.g., the contour integral method can be parallelized without modifying the internals of the package, as illustrated in the tutorials.

We provide natively implemented classes of NEPs, as well as gallery problems, including problems which are viewed as difficult (e.g. in the sense that they cannot be expressed as a short sum as in \((\ast)\)). Moreover, by using Julia’s interoperability features, we make it possible to convert the Berlin-Manchester benchmark problems [1] (including the updated problems in v4.0) to the NEP-PACK format such that the solvers and techniques of the package can be applied.

Data structures. The software is designed in an object oriented manner, where the fundamental NEP-class represents a general NEP, and specific types inherit from this class. Typical common

\(^{1}\)Our nleigs implementation is a line-by-line conversion of the publicly available MATLAB-implementation of NLEIGS of the authors of the nleigs paper.
types, such as those arising from time-delay systems, polynomials, rational functions are natively supported by the package. A generic type is provided for problems which can be expressed as a short sum of products of matrices and functions

\[ M(\lambda) = A_1 f_1(\lambda) + \cdots + A_m f_m(\lambda). \quad (\star) \]

A user, or algorithm developer, can access the data of a NEP by using a number of interface functions. Essentially, three procedures can be used to access a NEP-object:

- **compute_Mder**: computes a given derivative of \( M(\lambda) \).
- **compute_Mlincomb**: computes a linear combination of derivatives \( \sum_{\ell=0}^{k} a_\ell M^{(\ell)}(\lambda) x_\ell \), where \( v_\ell \in \mathbb{C}^n \) and \( a_\ell \in \mathbb{C}, \ell = 0, \ldots, k \). (The coefficients \( a_\ell \) are used to avoid IEEE-overflow.)
- **compute_MM**: computes the block residual, for (\( \star \)) defined as

\[ M(V, S) := A_1 V f_1(S) + \cdots + A_m V f_m(S), \]

where \( f_1, \ldots, f_m \), must be implemented in a matrix function sense.

Although the specific types and the generic type (\( \star \)) include many NEP applications, the package is not restricted to those types. Due to Julia’s multiple dispatch feature, a user with a completely different NEP can implement their own interface functions. Rather than forcing such a user to always implement all interface functions, which might require considerable software development, we provide general poor-mans-implementations via wrappers between the interfaces. In this way you do not need to implement all interface functions, unless it is really necessary for efficiency. For instance, given an implementation of **compute_MM**, the **compute_Mlincomb** function can be computed from the relation

\[
\sum_{\ell=0}^{k} a_\ell M^{(\ell)}(\lambda) v_\ell = a_0 (A_1 V f_1(S) + \cdots + A_m V f_m(S)) e_1 = a_0 M(V, S) e_1,
\]

and

\[
S := \begin{bmatrix}
\lambda & 1 \cdot a_1 / a_0 & \cdots & \cdots & \lambda \\
& \ddots & \ddots & \ddots & \ddots \\
& & \cdots & \cdots & \lambda \\
& & & \ddots & \lambda \\
& & & & \lambda
\end{bmatrix}^T.
\]

An additional feature of our software design allows for transparent implementations of transformations and manipulations of NEP-objects. Nonlinear Rayleigh-Ritz projection is implemented as a transformation that results in a new NEP-object (of much smaller dimension) and essentially any of the NEP-algorithms of the package can be used to solve the projected problem. Certain types of deflation has been implemented in an algorithm independent way. The technique described in [3] deflates eigenvalues by constructing a larger NEP, with a smaller eigenvalue set, which we implement as a procedure resulting in an instantiation of a new NEP-object, such that we can combine deflation for essentially any algorithm in the package. Shifting, scaling and Mbius transformation are also provided by manipulation of the NEP-object. Chebyshev interpolation is provided in a way that it transforms a general NEP, to a NEP of the type (\( \star \)), where \( f_1, \ldots, f_m \) are (scaled and shifted) Chebyshev polynomials, such that (again) most solvers of the package can be used.
References


Hybrid Projection Methods for Large-scale Inverse Problems with Mixed Gaussian Priors

Taewon Cho, Julianne Chung, Jiahua Jiang

Department of Mathematics, Virginia Tech

https://intranet.math.vt.edu/people/jiahua/home.html

Abstract

For many imaging systems, better image reconstructions can be obtained by including more sophisticated image priors. Oftentimes prior knowledge will come from a combination of sources, and striking a good balance of information is critical for the reconstruction. For example, priors may be learned from available training data, but bias in reconstructions can be a big concern (e.g., when the training set is small or the desired image is very different from training set). Thus, a safer approach is to include a prior that combines learned information with conventional smoothness properties. In other scenarios (e.g. in seismic tomography), the desired solution may consist of components with different smoothness properties, and the correct mixture of smoothness priors can be difficult to know a priori.

We address these challenges by considering inverse problems with mixed Gaussian priors, where the prior covariance matrix can be represented as a convex combination of matrices. Then, we develop hybrid iterative projection methods for the efficient computation of solutions, and we investigate tools for selecting the regularization parameter and the mixing parameter automatically.

We are interested in linear inverse problems of the form,

\[ d = As + \epsilon \]  

where \( d \in \mathbb{R}^m \) contains the observed data, \( A \in \mathbb{R}^{m \times n} \) models the forward process, \( s \in \mathbb{R}^n \) represents the desired parameters, and \( \epsilon \in \mathbb{R}^m \) represents noise in the data. We assume that \( \epsilon \sim \mathcal{N}(0, R) \), where \( R \) is a positive definite matrix whose inverse and square root are inexpensive (e.g., a diagonal matrix). The goal is to compute an approximation of \( s \), given \( d \) and \( A \).

Since \( A \) is ill-posed, a small error in the data may lead to large errors in the computed approximation of \( s \). Regularization is required to stabilize the inversion process. There are many types of regularization. Here we assume a prior for \( s \) and consider a Bayesian framework. More specifically, we treat \( s \) as a Gaussian random variable with mean \( \mu \in \mathbb{R}^n \) and covariance matrix \( Q \in \mathbb{R}^{n \times n} \). That is, \( s \sim \mathcal{N}(\mu, \lambda^{-2}Q) \), where \( \lambda \) is a scaling parameter (yet to be determined) for the precision matrix.

In many applications, the choice of \( Q \) is pre-determined (e.g., using expert knowledge) and is chosen to enforce smoothness or regularity conditions on the solution. However, in some cases, we don’t have enough information to determine \( Q \) completely. These scenarios motivate us to consider mixed Gaussian priors, where the covariance matrix is a convex combination of matrices. Without loss of generality, we assume two matrices and consider prior covariance matrices of the form,

\[ Q = \gamma Q_1 + (1 - \gamma) Q_2 \]  

where \( Q_1 \) is a symmetric positive definite matrix, \( Q_2 \) is a symmetric positive semi-definite matrix, and mixing parameter \( 0 < \gamma \leq 1 \). We consider the case where computing matrix-vector products with \( Q_1 \) is easy, but accessing \( Q_1^{-1} \) or its symmetric factorization (e.g. Cholesky or eigenvalue decomposition) is not possible. Such scenarios arise, for example, when the prior covariance matrix...
is modeled entry-wise using covariance kernels (e.g., those from the Matérn class) or when non-isotropic priors must be used. In such cases, the main challenge is that the resulting covariance matrix is large and dense, so factorizing or inverting it is not feasible. However, matrix-vector multiplications can be done efficiently. See [1] for various examples. The only assumption we have for \( Q_2 \) is that matrix-vector products with \( Q_2 \) can be computed efficiently. The choice of \( Q_2 \) is wide. For example, in modern imaging applications \( Q_2 \) can be sample covariance matrix obtained from training data.

Since the inverse and/or factorization of \( Q \) is not available, we exploit the generalized Golub-Kahan bidiagonalization [1]. Generalized hybrid iterative projection methods have enabled the efficient computation of solution estimates for linear inverse problems where it is impossible or undesirable to compute and store the prior covariance matrix. In this work, we develop a hybrid iterative projection method that is based on an augmented, generalized Golub-Kahan bidiagonalization to approximate the MAP estimate,

\[
\mathbf{s}_{\text{MAP}} = \arg \min_{\mathbf{s}} \frac{1}{2} \| \mathbf{A}\mathbf{s} - \mathbf{d} \|_{R^{-1}}^2 + \frac{\lambda^2}{2} \| \mathbf{s} - \mathbf{\mu} \|_{Q^{-1}}^2 .
\]

(3)

where \( Q \) is of the form (2). The proposed method has two distinctive features. First, we assume that both \( \gamma \) and \( \lambda \) are unknown a priori and must be estimated during the solution process. For problems where \( \gamma \) is fixed in advance, generalized hybrid methods can be directly applied. However, developing a method where \( \gamma \) can be selected adaptively is not an obvious extension since the solution subspace should not depend on the unknown parameters. Our approach follows a hybrid framework where the problem is projected onto Krylov subspaces of small but increasing dimension, and the regularization parameter and mixing parameter can be automatically selected. Second, we describe and investigate various scenarios where training data can be used to define \( Q_1 \) and \( Q_2 \), so our approach can be considered a learning approach for the regularization term.

References

Parallel Numerical Linear Algebra for Future Extreme Scale Systems: Challenges and Highlights of NLAFET

Bo Kågström and the NLAFET Team

Dept. of Computing Science and HPC2N, Umeå University, Sweden

Abstract

The NLAFET project, funded by the EU Horizon 2020 Research and Innovation Programme has formally ended recently. The NLAFET project has developed a new generation of computational tools and software for problems in numerical linear algebra; tools that will simplify the transition to the next generation of HPC architectures. NLAFET thus provides a software infrastructure that many leading-edge applications must have for attaining high performance on extreme-scale systems. Many of the methodologies, functionalities and solutions developed are applicable to the development of numerical solutions for a wide range of applications.

The work in the NLAFET project has focused on fundamental scientific issues and software development, including novel algorithms, prototype and library software for a critical set of fundamental linear algebra operations. The results are presented in 39 deliverables and 23 NLAFET Working Notes available on the NLAFET web-site (www.nlafet.eu/publications). In this presentation, we give a brief overview of the work performed including some challenges and highlights.

The work package Dense Linear Systems and Eigenvalue Problem Solvers has designed algorithms efficient, scalable, and robust parallel software for fundamental dense matrix computations far above and beyond those covered by standard packages today. This WP delivers novel results on:

- linear system solvers,
- BLAS for heterogeneous systems,
- non-symmetric standard and generalized eigenvalue problem solvers including robust eigenvector computation,
- singular value problem solvers.

The work package Direct Solution of Sparse Linear Systems has designed parallel algorithms and software for the direct solution of sparse linear equations. This WP delivers novel results on:

- lower bounds on communication for sparse matrices,
- direct solvers for (near-)symmetric systems,
- direct solvers for highly unsymmetric systems,
- hybrid direct-iterative solvers.

The work package Communication-Optimal Algorithms for Iterative Methods has designed new iterative methods that allow to drastically reduce the communication, and even minimize it whenever possible. This WP delivers novel results on:

- computational kernels for preconditioned iterative methods,
• iterative (enlarged Krylov) methods,
• multilevel preconditioners.

The work package **Cross-Cutting Issues** has investigated and designed a sustainable set of methods and tools for scheduling and runtime systems, auto-tuning, and algorithm-based fault tolerance (ABFT) packaged into open source library modules. This WP delivers novel results on:

• parallel critical path (PCP) scheduling for improving scalability; evaluation of several scheduling strategies for solving dense linear algebra problems on various HPC systems,
• scheduling strategies for solving dense linear algebra problems on various HPC systems,
• self-adaptive approaches for on-line tuning,
• implementation of a task-based ABFT Cholesky factorization.

The work package **Challenging Applications a Selection** has integrated and evaluated software parts of the NLAFET Library in the following applications:

• Task-based shared memory parallelism into 2DRMP software for modelling of electron scattering from H-like atoms and ions
• Load flow based calculations in large-scale power systems and PowerFactory code
• Communication avoiding iterative methods for solving linear systems arising from several different applications, in particular linear elasticity problems
• Data analysis in astrophysics and Midapack

One main achievement of the NLAFET project is the software developed and deployed, and made available via the NLAFET website http://www.nlafet.eu/software/. The public GitHub repositories that together constitute the NLAFET library software are structured in five groups:

• Dense matrix factorizations and solvers
• Solvers and tools for standard and generalized dense eigenvalue problems
• Sparse direct factorizations and solvers
• Communication optimal algorithms for iterative methods
• Cross-cutting tools

The parallel programming models used in various combinations are MPI, OpenMP, PaRSEC and StarPU.

The goal is to further develop, push, and deploy software into the scientific community to make it competitive on a world scale, and to contribute to standardisation efforts in the area.

The NLAFET Consortium consists of four partners: Umeå University (Bo Kågström, main Coordinator and UMU-PI), University of Manchester (Jack Dongarra, UNIMAN-PI), Institute National de Recherche en Informatique et en Automatique (Laura Grigori, INRIA-PI), and Science & Technology Facilities (Iain Duff, STFC-PI). For more information about the NLAFET Team and the project see the NLAFET website: www.nlafet.eu
Don’t Matricize, Tensorize: Tensor-tensor Products for Optimal Representation and Compression

Misha E. Kilmer, Lior Horesh, Haim Avron and Elizabeth Newman

Department of Mathematics
https://mkilme01.pages.tufts.edu/

Abstract

The use of higher order tensor representations (i.e. multiway array representations) is natural in science and engineering applications. For example, one might store discrete data in a format that is consistent with the grid labelling, so that the value of entry \((i, j, k)\) in a third order tensor might correspond to function value at a specific spatial grid point \((x_j, y_i)\) at time \(t_k\). Nevertheless, when it is time to analyze or compress that same data, the data is often first “matricized” (i.e. unrolled into matrix form), and linear algebraic tools are used to decompose, analyze, or compress the data.

We are primarily concerned with optimal data\(^1\) compression with a third (or higher) order tensor versus optimal compression of the same information oriented in matrix form. Indeed, recasting data from an order \(d−1\) tensor to an order \(d\) (or higher) tensor may be just the key to revealing latent structure that will allow for better compression. Consider, for example, a single vector \(v \in \mathbb{R}^{mn}\) (i.e. a first order tensor) which is actually the Kronecker product of two vectors, one of length \(n\) and one of length \(m\). In order to reveal the two vectors in the Kronecker form, one can reshape the data to an \(m \times n\) or \(n \times m\) matrix (i.e. a second order tensor), and observe that it is a rank-one matrix. From the rank-1 matrix factors, one obtains a pair of vectors needed to form \(v\). Thus the implicit storage cost of \(v\) is only \((m + n)\) floating point numbers, rather than \(mn\), a great savings if \(m\) and \(n\) are large.

To compare compressed tensor representations with compressed matrix representations, one must choose a specific tensor decomposition and metric. The tensor algebras proposed first by Kilmer and Martin [1] and extended by Kernfeld, et al [2] offer matrix algebra mimetic properties. In particular, the algebras admit tensor SVDs which resemble their matrix counterparts and randomized approximations to those tensor SVDs are also possible [3]. Moreover, much of the work in computing these tensor SVDs can be computed efficiently in parallel. Thus, we will consider truncated tensor SVDs under this tensor-tensor product formulation.

After introducing the class of tensor-tensor products and corresponding tensor decompositions, we show that an Eckart-Young optimality theorem under the Frobenius norm, first proposed in [1] for a particular class of tensor-tensor product, can be extended to an entire family of tensor-tensor products by capitalizing on framework from [2]. Since such optimality properties can be proven in both matrix and tensor-based algebras, the important question is whether either construct subsumes the other in terms of representation efficiency. We address this by proving that a Frobenius-norm optimal \(k\)-term tensor-tensor approximation is superior to its \(k\)-term truncated matrix SVD counterpart, and give additional optimal variants that improve this result. We highlight the implications of these results for such applications as image compression, model reduction, and matrix operator approximation. Time permitting, we also show how these truncated tensor-tensor product representations are related to other familiar tensor decompositions such as the truncated HOSVD.

\(^1\)Here, the term ‘data’ is loosely defined to mean any collection of numbers to which a multi-way ordering can be associated.
References


Order reduction methods for solving large-scale differential matrix Riccati equations

Gerhard Kirsten, and Valeria Simoncini

Dipartimento di Matematica, Università di Bologna
https://sites.google.com/view/gerhard-kirsten

Abstract

We consider the solution of the continuous-time differential matrix Riccati equation (DRE in short) of the form

\[ \dot{P}(t) = A^T P(t) + P(t) A - P(t) B B^T P(t) + C^T C, \quad P(t_f) = P_f. \]  

(1)

in the unknown matrix \( P(t) \in \mathbb{R}^{n \times n} \), where \( P_f = Z Z^T \) and \( t \in [0, t_f] \). Here, \( A, B \in \mathbb{R}^{n \times n} \), \( C \in \mathbb{R}^{p \times n} \), and \( Z \in \mathbb{R}^{n \times q} \) are time invariant, and \( s, p, q \ll n \). Under the considered hypotheses, numerical evidence seems to indicate that \( P(t) \) usually has rapidly decaying singular values, hence a low-rank approximation to \( P(t) \) may be considered.

The DRE plays a fundamental role in optimal control theory. In particular, equations of the form (1) are crucial in the numerical treatment of the linear quadratic regulator (LQR) problem [1, 2]: given the state equation

\[ \dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t), \quad x(0) = x_0 \]  

(2)

the problem is to find a feedback optimal control \( \tilde{u}(t) \) that minimizes a cost integral. Consider the finite horizon case, where the finite time cost integral has the form

\[ J(u) = x(t_f)^T P_f x(t_f) + \int_0^{t_f} (x(t)^T C^T C x(t) + u(t)^T u(t)) \, dt. \]  

(3)

Assuming that the pair \((A, B)\) is stabilizable and the pair \((C, A)\) is detectable, the optimal input \( \tilde{u}(t) \), minimizing (3), can be determined as \( \tilde{u}(t) = -B^T P(t) \tilde{x}(t) \), and the optimal trajectory is subject to \( \dot{\tilde{x}} = (A - BB^T P(t))x(t) \). The matrix \( P(t) \) is the solution to the DRE (1).

In the framework of differential equations, the DRE is characterized by both fast and slowly varying modes, hence it is classified as a stiff ordinary differential equation (ODE). The stiffness and the nonlinearity of the DRE are responsible for the difficulties in its numerical solution even on a small scale \( (n < 10^3) \). On a large-scale these challenges are particularly evident and, hence, the efficient solution of large-scale DREs has been addressed in the recent literature. Recent algorithms have focused on first discretizing the DRE in time, after which approximation methods are used at each timestep to solve the resulting algebraic equation \([3, 4]\). These methodologies may lead to high computational costs and memory requirements. To avoid this, a promising idea is to rely on an order reduction strategy typically used in linear and nonlinear dynamical systems. In this setting, the original system is replaced with

\[ \dot{\tilde{x}}(t) = A_m \tilde{x}(t) + B_m u(t), \quad y(t) = C_m \tilde{x}(t), \quad \tilde{x}(0) = \tilde{x}_0 \]  

(4)

where \( A_m, B_m \) and \( C_m \) are projections and restrictions of the original matrices onto a subspace of small dimension. The dimension of the original DRE is thus reduced once and for all, after which time discretization is performed in the space of reduced dimension. This strategy allows for
a natural low-rank approximation to the sought after DRE solution \( P(t) \), obtained by interpolating the reduced order solution at selected time instances.

A key ingredient for the success of the reduction methodology is the choice of the approximation space onto which the algebraic reduction is performed. The authors of [5] and [6] have independently used polynomial and extended Krylov subspaces as approximation space, respectively. In this presentation we show that great computational and memory savings are obtained by a reduction process onto rational Krylov subspaces. A related key issue is the expected final accuracy and thus the stopping criterion. Time dependence of the DRE makes the reduced problem trickier to handle than in the purely algebraic case; in particular, two intertwined issues arise:

i) The approximate solution accuracy may vary considerably within the time interval \([0, t_f]\);

ii) Throughout the reduction process, the reduced ODE needs to be solved with low-order methods to make the overall cost feasible.

In this presentation we address these specific difficulties by deriving a new stopping criterion that takes into account the different approximation behavior of the algebraic and differential portions of the problem, together with a refinement procedure that improves the final approximate solution by using a high-order integrator. The new method allows us to numerically solve much larger problems than in the current literature. We illustrate the efficiency in terms of storage and computational time compared to existing methods on benchmark numerical examples [7].

The outlook is to generalize our methodology to more general nonlinear, parameter dependent differential matrix equations. In particular, we are interested in systems of the form

\[
\dot{P}(t, \mu) = AP(t, \mu) + P(t, \mu)A^T + N(P, t, \mu) + C^T C, \quad P(0, \mu) = ZZ^T, \tag{5}
\]

with matrix dimensions similar to (1). Here \( N(P, t, \mu) \) is a nonlinear, sufficiently regular matrix function, depending on \( P \) but also on some vector of parameters \( \mu \in \mathbb{D} \subset \mathbb{R}^\ell, \ell = 1, 2 \ldots \), as well as time \( t \in [0, t_f] \). It is well known that the challenge posed by this type of problem is the efficient handling of the nonlinear term when the problem has large dimension and many parameters need to be analyzed. To attack the nonlinear term, we aim to devise matrix-oriented variants of the discrete empirical interpolation method (DEIM) [8].

References


A multivariate Crouzeix-Palencia bound and its applications

Daniel Kressner

Institute of Mathematics, École Polytechnique Fédérale de Lausanne, Switzerland
https://anchp.epfl.ch

Abstract

Consider a function $f : \Omega \to \mathbb{C}$ analytic on a domain $\Omega \subset \mathbb{C}$ containing the closure of the numerical range

$$W(A) := \{ v^* A v : v \in \mathbb{C}^n, \|v\|_2 = 1 \}$$

of a matrix $A \in \mathbb{C}^{n \times n}$, where $\| \cdot \|_2$ denotes the Euclidean norm for vectors and the spectral norm for matrices. The celebrated result by Crouzeix and Palencia [3] states that

$$\|f(A)\|_2 \leq (1 + \sqrt{2}) \sup_{z \in W(A)} |f(z)|.$$  \hspace{1cm} (1)

The purpose of this talk is to discuss an extension of (1) to multivariate functions and its implications. The notion of multivariate matrix functions considered here follows [5], which is a special case of the well established holomorphic functional calculus on commutative Banach algebras. For example, for $p(x,y) = 1 + xy + x^3y^2$ and matrices $A \in \mathbb{C}^{m \times m}, B \in \mathbb{C}^{n \times n}$, we set

$$p\{A,B\} = I + B \otimes A + B^2 \otimes A^3 \in \mathbb{C}^{mn \times mn},$$

where $\otimes$ denotes the usual Kronecker product. Equivalently, $p\{A,B\}$ can be viewed as the linear operator on $\mathbb{C}^{m \times n}$ defined by $p\{A,B\} : X \mapsto X + AXB^T + A^3X(B^T)^2$. For a general bivariate function $f$ analytic on a domain $\Omega \subset \mathbb{C} \times \mathbb{C}$ containing the Cartesian product of the eigenvalues of $A$ and $B$, we define

$$f\{A,B\} := -\frac{1}{4\pi^2} \int_{\Gamma_A} \int_{\Gamma_B} f(x,y) (yI - B)^{-1} \otimes (xI - A)^{-1} \, dy \, dx,$$  \hspace{1cm} (2)

for closed contours $\Gamma_A$ and $\Gamma_B$ enclosing the eigenvalues of $A$ and $B$, respectively, and satisfying $\Gamma_A \times \Gamma_B \subset \Omega$. This definition includes, when $f(x,y) = 1/(x - y)$, the solution operator for a Sylvester matrix equation and, when $f$ is the finite difference quotient, the Fréchet derivative of a (univariate) matrix function.

Assuming that $f$ is analytic on a domain $\Omega$ containing $\overline{W(A)} \times \overline{W(B)}$, our main result shows that

$$\|f\{A,B\}\|_2 \leq (1 + \sqrt{2})^2 \|f\|_{W(A) \times W(B)},$$  \hspace{1cm} (3)

where $\|f\|_{W(A) \times W(B)}$ denotes the supremum of $|f|$ on $W(A) \times W(B)$. As far as we are aware of, a result by Starke [8] for $f(x,y) = 1/(x - y)$ is the only previously known instance for a result of the form (3). Our proof of (3) makes use of the matrix-valued variant of (1) already pointed out in [3] and the simplified techniques described in [7].

The definition and the result (3) extend to analytic functions in $d \geq 3$ variables. In this case, the constant becomes $(1 + \sqrt{2})^d$. Clearly, these constants are worse than in (1) but a simple example shows that the constant can in fact not be smaller than $2^d$. In particular, the exponential growth with respect to $d$ is unavoidable.
Applications and special cases of (3) include:

- A bound for the norm of the Fréchet derivative $Df(A)$ of a univariate function $f$ at a matrix $A$:
  \[ \|Df(A)\|_2 \leq (1 + \sqrt{2})^2 \sup_{z \in W(A)} |f'(z)|. \]

  This is of interest because $\|Df(A)\|_2$ governs the conditioning of $f(A)$ and, as far as we know, no such bound was previously known.


- A greatly simplified convergence analysis of the Krylov subspace method presented in [1] for performing low-rank updates of matrix functions.

This talk is partly based on joint work in preparation with Michel Crouzeix, Université de Rennes.

References


Admissible and attainable convergence behavior of block Arnoldi and GMRES

Marie Kubínová and Kirk M. Soodhalter

The Czech Academy of Sciences, Institute of Geonics
http://www.ugn.cas.cz/~kubinova/

Abstract

It was shown in [3] that any non-increasing convergence curve is possible for GMRES and a family of pairs \((A, b)\) can be constructed for which GMRES exhibits a given convergence curve with \(A\) having arbitrary spectrum. However, so far no analogue of this result has been established for block GMRES (blGMRES), wherein multiple systems with the same coefficient matrix are considered, i.e.,

\[
AX = B, \quad A \in \mathbb{C}^{m \times m}, \quad B \in \mathbb{C}^{m \times s}. \tag{1}
\]

We explore if blGMRES residual convergence admits a similar characterization as in [3]. Among others, we would like to answer the following question: If for one given right-hand side we can construct a matrix that leads to an arbitrarily fast/slow GMRES convergence, can we achieve the same for two or more right-hand sides simultaneously?

Framework

Most often, analysis of blGMRES takes the view of the method as a minimization of each individual residual over a sum of spaces, i.e., a method treating a collection of scalar linear systems. To capture the full picture of block methods behavior, which is influenced by the interaction between the different right-hand sides, it is advantageous to formulate these methods in terms of vector blocks in \(\mathbb{C}^{m \times s}\), as in [2] and [5]. To this end, we define a block analogue of the inner product as

\[
\langle\langle X, Y \rangle\rangle := Y^* X \in \mathbb{S} \simeq \mathbb{C}^{s \times s} \tag{2}
\]

and formulate the results through the \(*\)-algebra \(\mathbb{S}\) of \(s\)-by-\(s\) complex matrices. Embracing this totally block view greatly simplifies analysis of block methods and allows us to obtain clean convergence results.

Admissible convergence behavior

In the standard case, an admissible residual convergence curve is defined as a sequence of non-negative non-increasing numbers. It is however not straightforward how this definition translates to the block case. It cannot be simply multiple independent non-increasing sequences, it has to take into account the relation between the right-hand sides, since if two individual right-hand sides are almost linearly dependent, one cannot expect radically different convergence behaviors for each of them.

The residual convergence of a norm-minimizing method can be described through the peak-plateau relationship with its Galerkin counterpart. The block inner product (2) allows to generalize the peak-plateau relation to block methods. In particular, whenever the block FOM (blFOM) solution exist, the residuals of blGMRES and blFOM satisfy

\[
\langle\langle R_k^F, R_k^F \rangle\rangle^{-1} = \langle\langle R_k^G, R_k^G \rangle\rangle^{-1} - \langle\langle R_{k-1}^G, R_{k-1}^G \rangle\rangle^{-1}. \tag{3}
\]
Although some relations between blFOM and blGMRES can be obtained also using standard linear algebra tools, such as in [6], with the new framework, the analysis is more elegant and shadows that of the standard GMRES and FOM.

Relation (3) has some nontrivial consequences for the convergence behavior of blGMRES. In particular, if there is no breakdown in the block Arnoldi algorithm, the blGMRES residuals satisfy

\[
\langle R_0, R_0 \rangle \overset{\text{Loewner}}{\preceq} \langle R_1^G, R_1^G \rangle \overset{\text{Loewner}}{\preceq} \cdots \overset{\text{Loewner}}{\preceq} \langle R_{n-1}^G, R_{n-1}^G \rangle \overset{\text{Loewner}}{\succ} 0,
\]

where \(\prec\) and \(\preceq\) is the Loewner (partial) ordering of Hermitian matrices. Relation (4) trivially implies monotonic convergence of the size of the individual residuals, but it is generally stronger, because it takes into account the inter-residual relationships stored in the off-diagonal entries of the matrices \(\langle R_k^G, R_k^G \rangle\).

Attaining prescribed convergence behavior

We utilize the new framework for block Krylov subspace methods and generalize some of the results of [1, 3] valid for standard Arnoldi and GMRES to the block case to show that any convergence behavior satisfying (4) is attainable by blGMRES. Moreover, under moderate conditions, we can also simultaneously prescribe the spectral properties of \(A\) and the principal submatrices \(H^{(k)}\) of the ultimate upper Hessenberg matrix \(H\) produced by blArnoldi\((A, B)\). These entities will be however prescribed in a slightly different form than in the standard case.

In the standard case, the spectral properties of the coefficient matrix are prescribed through annihilation of the characteristic polynomial. In the block case, we can analogously require \(A\) and \(B\) to annihilate a given polynomial \(M\) with matrix-valued coefficients \(C_{n-1}, \ldots, C_0 \in \mathbb{S}\), i.e.,

\[
M(A) \circ B := A^n B - \sum_{k=0}^{n-1} A^k B C_k = 0,
\]

\(C_0\) nonsingular.

In the contribution, we will further discuss the relation between the spectral properties of \(A\) and \(H^{(k)}\) and the polynomials with matrix-valued coefficients they annihilate. We will provide a complete characterization of pairs \((A, B)\) giving the prescribed convergence behavior of blArnoldi as well as blGMRES, as given in [4]. Since enforcing (5) is even more restrictive than prescribing solely the eigenvalues of \(A\), we conclude that the spectrum is even less indicative with respect to the residual convergence than in the standard case. At the end, we will also present results related to blGMRES with restarts.

References


Numerical linear and MULTILINEAR ALGEBRA methods for systems of POLYNOMIALS

Patrick Kürschner, Lieven De Lathauwer

KU Leuven, ESAT/STADIUS & Campus KULAK Kortrijk, https://sites.google.com/view/patrickkuerschner

Abstract

We consider the numerical solution of systems of nonlinear equations in $n$ variables,

$$f_1 = \ldots = f_s = 0,$$

where the $f_i = f_i(x_1, \ldots, x_n)$, $i = 1 : s$, are $n$-variate polynomials of degree $d > 1$. This setting naturally extends linear systems of equations, $Ax = b$. Applications of such polynomial systems of equations include, for instance, signal processing, system identification, machine learning, uncertainty quantification, and multivariate interpolation and optimization.

In the first part of the talk we introduce the Macaulay matrix framework [3] which allows one to tackle the nonlinear rootfinding problem by means of techniques from numerical linear algebra, such as nullspace and eigenvalue computations. There, the original polynomial system is expanded by adding extra polynomial equations derived from all $f_i$ and collecting all the resulting coefficients in a matrix $M$, referred to as the Macaulay matrix. The nullspace of the Macaulay matrix encodes all the information of the solution structure of the system. From a numerically computed nullspace matrix $K$, $MK = 0$, one can retrieve all roots of the polynomial system by means of eigenvalue problems [2].

We present a recent alternative for the root retrieval, where the numerically computed nullspace basis $K$ is reshaped into a third-order tensor $T$. This tensor admits low-rank tensor decompositions in the form of canonical polyadic or block-term decompositions [4, 5], whose low-rank factors then reveal the roots. This tensor based root retrieval has several advantages such as the ability to work with a Macaulay matrix of smaller size and to tackle also overconstrained systems (i.e., systems with more equations than variables, $s > n$).

The second part of the talk is concerned with the computational bottleneck in both the eigenvalue and tensor based root retrieval: the computation of a numerical basis $K$ for the nullspace of the Macaulay matrix $M$. For increasing values of $n$, $d$, the Macaulay matrix quickly becomes very large.

For example, a harmless appearing system of quadratic polynomials ($d = 2$) in $n = 10$ variables leads to a Macaulay matrix of dimensions $437580 \times 184756$.

Consequently, computing the nullspace of $M$ becomes a very demanding task. We investigate the use of algorithms for large-scale eigenvalue or singular value problems for this purpose. By viewing the multivariate polynomials as multilinear forms, we show that the Macaulay matrix represents a flattening of a higher order tensor. This multilinear algebra point of view reveals new insights into the structure of the Macaulay matrix and, moreover, can be exploited to efficiently compute matrix vector products and to derive preconditioning strategies for the eigen- or singular value methods.

Additionally, every vector in the nullspace can be expressed by a low-rank tensor factorization, e.g., a polyadic decomposition. This can be used to substantially reduce the memory requirements in both the nullspace computation and the root retrieval. Moreover, the nullspace problem $MK = 0$ can be seen as a linear system of equations where the solution $K$ is a flattening of a canonical polyadic decomposition, for which specialized, optimization based routines [1] can be used.
References


Delaying the second orthogonalization in Classical Gram-Schmidt with Reorthogonalization to obtain a one-synchronization Gram-Schmidt algorithm for QR factorizations and Arnoldi expansions. Application to GMRES and Krylov-Schur algorithms.

Julien Langou, Kasia Świrydowicz, Stephen Thomas, Daniel Bielich
Mathematics Department, University of Colorado, Denver
http://www-math.ucdenver.edu/~langou/index.html

Abstract

We are interested in the Gram-Schmidt orthogonalization scheme in the context of an Arnoldi expansion where columns are added one at a time. For each column, (at each step,) an orthogonal projection and a normalization are performed.

Our goal is to reduce the number of synchronization steps in the algorithm. To do so, our method is to group inner products. Indeed, a group of inner products can be performed in a common synchronization step. The term synchronization comes from the parallel distributed world where processors need to get in synchronization in order to compute the inner product through a global reduction.

In this context, we present an Arnoldi expansion algorithm that is fully stable and requires only one synchronization per step.

In 2018, Świrydowicz et al. [2] present a collection of Gram-Schmidt orthogonalization with one synchronization per step. However these methods lose orthogonality as $O(\varepsilon k)$. In practice, such a loss of orthogonality is innoxious to GMRES. However, such a loss of orthogonality seriously impairs Arnoldi to find eigenvalues. For good performance with Arnoldi, we need an orthogonalization scheme with an $O(\varepsilon)$ loss of orthogonality.

In order to obtain a reasonable orthogonality, (so as to enable the Arnoldi algorithm to find eigenpairs,) one projection in Gram-Schmidt is not enough. We must use at least two projection steps. Two projection steps implies two synchronizations per column ... unless ... one is ready to delay the second projection to the next step. At the next step, we will then perform the first projection of the current vector (as usual) but with an “only once orthogonalized“ previous vector (instead of twice, so the previous vector is partially orthogonalized), and simultaneously we perform the second orthogonalization of the previous vector. Another way to describe the algorithm is to say: “we delay the second projection to the next Arnoldi step.”

We present the idea of delaying the second orthogonalization in the context of a QR factorization. See Algorithm . And then in the context of an Arnoldi expansion. See Algorithm . (There are some significant difficulties in Arnoldi expansion not present in QR factorization.)

In 2007, Hernandez et al. [1] present a similar idea. Their DCGS2 algorithm also delays the second orthogonalization to the next step, so that the algorithm performs one synchronization per columns. However the numerical impacts of the delayed second orthogonalization have not been taking into account resulting in an unstable algorithm. Delaying is not enough. One has to a-posteriori correct the quantities obtained by the “only once orthogonalized“ previous vector to transform them to the quantities that would have been obtained by the “projected twice“ previous vector.

Since the current vector is projected onto the “only once orthogonalized“ previous vector, (as opposed to the twice-orthogonalized previous vector,) corrections on the projection coefficients
need to be made. This is shown in line 5 of Algorithm in the context of a QR factorization. Namely the correction is $r_{j-1,j} = r_{j-1,j} - w^T z$. We call this Stephen’s trick.

Similarly in the context of Arnoldi, corrections need to be made. In this case there are two impacts. First we do need to use Stephen’s trick as in the context of a QR factorization. This is shown in line 15 of Algorithm in the context of a QR factorization. Namely the correction is $r_{k} - r_{1:k-1}^T t_{1:k-1}$. Second, and this is specific to Arnoldi, since the matrix-vector product is performed on the “only once orthogonalized” previous vector, the matrix-vector product is wrong. We should have done the matrix-vector product on the twice-orthogonalized previous vector But we did not have it, since we delayed. So what we need to do is correct again. This is shown in line 11 of Algorithm in the context of a QR factorization. Namely the correction is $H_t = H_{1:k,1:k-1} t_{1:k-1}$.

Finally, we note that synchronization also comes from the normalization step. In this abstract, we have not discussed the synchronization required in the normalization step. They can be tackled by delaying the normalization. The code in Algorithm and Algorithm show delayed normalizations.

The QR factorization algorithm (Algorithm ) experimentally present perfect stability and has only one synchronization per column added. The Arnoldi expansion algorithm (Algorithm ) experimentally present perfect stability and has only one synchronization per column added.

Our work comes with a full test suite and software library. Demonstration codes in Matlab and C and C+MPI are available. Many orthogonalization methods are provided for comparison: many Gram-Schmidt variants (in particular the ones presented in [2]), but also Householder reflection based codes. Examples with the Krylov Schur algorithm are used.

An extensive stability experimentation is performed and presented. It demonstrates the stability of the new methods. Performance results on large scale computational machine are presented too.

\textbf{Algorithm 1} Classical Gram-Schmidt (CGS-2) Algorithm with Normalization Lag and Reorthogonalization Lag

\textbf{Input:} Matrices $Q_{j-1}$ and $R_{j-1}$, $A_{j-1} = Q_{j-1} R_{j-1}$; column vector $q_j = a_j$

\textbf{Output:} $Q_j$ and $R_j$, such that $A_j = Q_j R_j$

1: if $j = 1$ return
2: $[r_{j-1,j-1}, r_{j-1,j}] = q_j^T [q_{j-1}, q_j]$ \Comment{Global synch}
3: if $j > 2$ then
4: $[w, z] = Q_{j-2} [q_{j-1}, q_j]$ \Comment{same global synch}
5: $[r_{j-1,j-1}, r_{j-1,j}] = [r_{j-1,j-1} - w^T w, r_{j-1,j} - w^T z]$ \Comment{Lagged $R$ update}
6: $r_{1:j-2,j-1} = r_{1:j-2,j-1} + w$
7: end if
8: $r_{j-1,j-1} = \{r_{j-1,j-1}\}^{1/2}$ \Comment{Lagged norm}
9: if $j > 2$ $q_{j-1} = q_{j-1} - Q_{j-2} w$ \Comment{Lagged Reorthogonalization}
10: $q_{j-1} = q_{j-1} / r_{j-1,j-1}$ \Comment{Lagged Normalization}
11: $r_{j-1,j} = r_{j-1,j} / r_{j-1,j-1}$ \Comment{Apply recursive projection}
12: $r_{1:j-2,j} = z$
13: $q_j = q_j - Q_{j-1} r_j$

\textbf{Algorithm 2} Arnoldi Classical Gram-Schmidt (Arnoldi CGS-2) with Normalization Lag and Reorthogonalization Lag
\begin{verbatim}
1: \begin{align*}
q_1 &= \frac{r_1}{\|r_1\|}, \\
h_{1,1} &= q_1^T A q_1; q_2 = A q_1 - h_{1,1} q_2 \\
\text{for } k = 2, 3, \ldots \text{ do} \\
y &= A q_k \\
[t r] &= [Q_k^T q_k \ Q_k^T y] \\
q_k &= q_k - Q_{k-1} \ast t_{1:k-1} \\
nrm &= \sqrt{r_k - t_{1:k-1}^T t_{1:k-1}} \\
H_{k,k-1} &= nrm \\
q_k &= q_k / nrm \\
H_{1:k,k-1} &= H_{1:k-1,k-1} + t_{1:k-1} \\
C &= Q_{k-2}^T Q_{k-1}, \quad H_t = H_{1:k,1:k-1} t_{1:k-1} \\
p &= [t_{1:k-1} \quad (t_k - t_{1:k-1}^T t_{1:k-1}) / nrm] \\
H_{1:k-1,k} &= r_{1:k-1} - C H_t, \quad H_{1:k-1,k} = H_{1:k-1,k} / nrm \\
H_{k,k} &= r(k) - t_{1:k-1}^T r_{1:k-1} - p H_t + t_{1:k-1}^T C H_t \\
p &= [r_{1:k-1} \quad (r_{k} - r_{1:k-1}^T t_{1:k-1}) / nrm]; \\
q_{k+1} &= y - Q_k p \\
\end{align*}
\end{verbatim}

\textbf{References}


Abstract

Photonic crystals are made up of position-dependent dielectric materials with periodic structures. Full band gap is the most distinguished feature of photonic crystals and attracts extensive studies in its properties and applications. Numerical simulations play an essential role to predict the band structures with various geometric settings, which purposefully affect the electromagnetic wave propagation.

Mathematically, the propagation of electromagnetic fields is modelled by the three-dimensional (3D) frequency domain source-free Maxwell’s equations with the constitutive relations

\[
\begin{align*}
\nabla \times \mathbf{E}(x) &= \omega \mathbf{B}(x), \\
\nabla \times \mathbf{H}(x) &= -\omega \mathbf{D}(x), \\
\n\nabla \cdot \mathbf{B}(x) &= 0, \\
\n\nabla \cdot \mathbf{D}(x) &= 0,
\end{align*}
\]

where \( \omega \) is the frequency, \( \mathbf{E}, \mathbf{H}, \mathbf{D} \) and \( \mathbf{B} \) are the electric, the magnetic fields, the dielectric displacement and the magnetic induction, respectively, at the position \( x \in \mathbb{R}^3 \). In linear dielectrics, the electromagnetic field satisfies \( \mathbf{B} = \mu \mathbf{H}, \mathbf{D} = \varepsilon \mathbf{E} \), where \( \mu \) is the permeability, \( \varepsilon \) is the permittivity. By eliminating the magnetic field, (1) becomes the differential Maxwell eigenvalue problem:

\[
\nabla \times \nabla \times \mathbf{E} = \omega^2 \mu \varepsilon \mathbf{E}.
\]

In chiral media, the coupling effects between the electromagnetic field can be described by the constitutive relations \( \mathbf{B} = \mu \mathbf{H} + \zeta \mathbf{E}, \mathbf{D} = \varepsilon \mathbf{E} + \xi \mathbf{H} \), where \( \zeta \) and \( \xi \) are magnetoelectric parameters. As a result, the corresponding Maxwell’s equations (1) can be rewritten as the differential Maxwell eigenvalue problem:

\[
\begin{bmatrix}
0 \\
\iota \nabla \times \\
0
\end{bmatrix}
\begin{bmatrix}
\mathbf{H} \\
\mathbf{E}
\end{bmatrix} = \omega
\begin{bmatrix}
\mu & \zeta \\
\xi & \varepsilon
\end{bmatrix}
\begin{bmatrix}
\mathbf{H} \\
\mathbf{E}
\end{bmatrix}.
\]

Based on the Bloch Theorem, the eigenvectors \( \mathbf{E} \) and \( \mathbf{H} \) in (2) and (3) on a given crystal lattice, satisfying the quasi-periodic conditions

\[
\mathbf{E}(x + \mathbf{a}_\ell) = e^{i2\pi k \cdot a_\ell} \mathbf{E}(x), \quad \mathbf{H}(x + \mathbf{a}_\ell) = e^{i2\pi k \cdot a_\ell} \mathbf{H}(x), \quad \ell = 1, 2, 3,
\]

where \( \{\mathbf{a}_\ell\}_{\ell=1}^3 \) are lattice translation vectors and \( 2\pi k \) is the Bloch wave vector within the first Brillouin zone.

Given a specific 3D photonic crystal, it can be proved that only certain nonzero real \( \omega \)'s can satisfy (1) simultaneously. Using Yee’s finite difference scheme [1] on (2) and (3) satisfying the source-free conditions and the quasi-periodic conditions (4), the discretized Maxwell eigenvalue problems (MEPs) result in the generalized eigenvalue problems

\[
Ax = \lambda Bx
\]

with

\[
A = C^H C \quad \text{and} \quad A = \begin{bmatrix}
0 & -\iota C \\
\iota C^H & 0
\end{bmatrix}, \quad B = \begin{bmatrix}
\mu_d & \zeta_d \\
\xi_d & \varepsilon_d
\end{bmatrix}
\]
for linear dielectrics (2) and chiral media (3), respectively, where $C$ has the special structure for the discrete curl operator $\nabla \times$ which can easily be treated with the fast Fourier transform (FFT) to accelerate the numerical simulation \[2\] with

$$
C = \begin{bmatrix}
0 & -C_3 & C_2 \\
C_3 & 0 & -C_1 \\
-C_2 & C_1 & 0
\end{bmatrix}
$$

(7)

and $C_1, C_2, C_3$ being the discretization forms of $\partial_x, \partial_y, \partial_z$ with Yee’s scheme.

Calculation of band structures of three dimensional photonic crystals amounts to solving a few eigenvalues with smallest magnitude of the large-scale Maxwell eigenvalue problems (5), which are notoriously challenging due to high multiplicity of zero eigenvalues because of the degenerate matrix $C$ in (6) and (7).

In this work, we try to address this problem in such a broad context that band structures of three dimensional isotropic photonic crystals in all 14 Bravais lattices can be efficiently computed in a unified framework. We uncover the delicate machinery behind several key results of our framework and on the basis of this new understanding we drastically simplify the derivations, proofs and arguments. Particular effort is made on reformulating the Bloch condition (4) for all 14 Bravais lattices in the redefined orthogonal coordinate system, and establishing the eigen-decompositions for discrete partial derivative operators in (7) by identifying the hierarchical structure of the underlying normal (block) companion matrix, and exploring an explicit form of orthogonal basis for the range and null spaces of matrix $C$ which includes diagonal matrices and FFT base matrix, thus contributes to a fast eigensolver.

To solve the MEPs (5) efficiently, we apply these theoretical results to project them to some null space free eigenvalue problems, which involve only the eigenspace associated with the nonzero eigenvalues, and therefore the zero eigenvalues are excluded and will not degrade the computational efficiency. With the validity of the novel nullspace free method in the broad context, we combine the inexact shift-invert residual Arnoldi method, invert Lanczos method and MINRES or PCG linear solver to solve the MEPs with a matrix dimension as large as 5,308,416. We perform some calculations on some benchmark systems to demonstrate the accuracy and efficiency of our algorithm. Related algorithm frameworks and numerical results have been shown on the website [3].

Especially for chiral media, the eigenstructure is heavily determined by the chirality parameter $\gamma$ in (3). We prove that all the eigenvalues are real and finite for a small chirality $\gamma$. For a critical value $\gamma = \gamma^*$, the MEP has $2 \times 2$ Jordan blocks at infinity eigenvalues. Numerical results demonstrate that when $\gamma$ increases from $\gamma^*$, the $2 \times 2$ Jordan block will first split into a complex conjugate eigenpair, then rapidly collide with the real axis and bifurcate into positive (resonance) and negative eigenvalues with modulus smaller than the other existing positive eigenvalues. The resonance band also exhibits an anticrossing interaction. Moreover, the electric and magnetic fields of the resonance modes are localized inside the structure, with only a slight amount of field leaking into the background (dielectric) material.

References


[3] https://sites.google.com/g2.nctu.edu.tw/fame/
A communication-avoiding 3D sparse triangular solver

Piyush Sao, Ramakrishnan Kannan, Xiaoye S. Li, Richard Vuduc

Computational Research Division, Lawrence Berkeley National Laboratory
https://portal.nersc.gov/project/sparse/xiaoye-web/

Abstract

We present a novel parallel algorithm for solving a sparse triangular system of linear equations, \( Tx = b \), where \( T \) is either an upper- or lower-triangular sparse matrix. A sparse triangular solver (SpTrs) is an important sub-step during LU and Cholesky factorization, which are direct methods for solving general linear systems. SpTrs also appears in preconditioners based on incomplete factorization, which commonly appear in Krylov subspace-based iterative methods.

The challenge is that the strong-scalability of prior state-of-the-art methods for performing this SpTrs is fundamentally limited by the cost of communication, because the arithmetic intensity is very low, about \( O(1) \).

In our previous work, we developed a communication-avoiding algorithm for LU factorization [1]. The idea underlying this 3D method is to organize the MPI processes logically into a three-dimensional grid, rather than a traditional 2D one, and then exploit the structure of the elimination tree—an abstraction that captures the data dependencies in sparse LU factorization—to replicate data judiciously. This combination of techniques provably reduces communication asymptotically in the problem size in common cases.

In this work, we leverage the 3D sparse LU data structure to develop a communication-avoiding SpTrs, which yields asymptotic reductions in the latency and communication-volume costs of a conventional SpTrs. Briefly, our new 3D SpTrs works as follows. Consider the 3D process grid as a collection of 2D MPI process grids. The prior technique of 3D factorization mapped independent subtrees of the elimination tree to each 2D process grid and replicated the common ancestors. Our 3D triangular solver exploits this same 3D organization. It first solves independent subtrees on different 2D process grids, and then performs a reduction before solving the subproblem in the common ancestor tree on a single 2D grid.

To analyze the communication and latency costs of our new method, we consider prototypical matrices arising from the discretization of “planar” and “non-planar” partial differential equations (PDEs). By planar, we mean the physical geometry of the input domain, when discretized, is flat or nearly so; we use the term planar instead of 2D to distinguish the problem geometry from that of the logical MPI process grid. Our analysis shows that the 3D SpTrs reduce the per-process communication volume asymptotically by a factor of \( O(n^{1/4}) \) and \( O(n^{1/6}) \) for problems arising from the finite element discretizations of 2D “planar” and 3D “non-planar” PDEs, respectively. This advantage comes at the cost of a small amount of additional memory needed to replicate the right-hand side.

We present empirical scalability results for our 3D SpTrs on up to 24k cores of a Cray XC30 machine. For a single right-hand side, our 3D SpTrs achieves a 4.6× and 1.8× speedup over the baseline 2D algorithm for planar and non-planar matrices, respectively. For multiple right-hand sides, our 3D SpTrs achieves 7.2× and 2.7× speed-up over the baseline 2D algorithm for planar and non-planar matrices, respectively [2].
References


The Forsythe Conjecture

Jörg Liesen

Institute of Mathematics, TU Berlin

www.math.tu-berlin.de/~liesen

Abstract

This presentation will be devoted to a conjecture of George Forsythe about the convergence behavior of certain iterative methods based on Krylov subspaces, which still remains unresolved after more than 50 years.

In his paper from 1968 [4], Forsythe studied the asymptotic behavior of the optimum $s$-gradient method for minimizing positive definite quadratic functions on $\mathbb{R}^n$. For simplicity he chose the model function $f(x) = \frac{1}{2}x^T Ax$, where $A \in \mathbb{R}^{n \times n}$ is symmetric positive definite. Since the unique minimizer of $f$ is the zero vector, the iterates $x_k$ coincide with the errors of the method. Computing $x_{k+1}$ from $x_k$ by the optimum $s$-gradient method corresponds to taking $s$ steps of the conjugate gradient (CG) method, starting from $x_k$. Thus, Forsythe was interested in the behavior of the CG method when it is restarted every $s$ steps. It is easy to show that the errors converge to the unique minimizer of $f$, but the interesting question is how the errors approach their limit.

For the case $s = 1$, which corresponds to the steepest descent method, Forsythe and Motzkin had conjectured already in 1951 that the errors with odd and even indices alternate asymptotically between two fixed directions that are determined by eigenvectors of $A$ belonging to its largest and smallest eigenvalues. In the terminology established by Stiefel in 1952 [7], these vectors form a “cage” (“Käfig”), which is an undesirable property since it means that the convergence of the method can be no faster than linear. The Forsythe-Motzkin conjecture was proved by Akaike in 1959 [2] using methods from probability theory, and by Forsythe in 1968 [4] using properties of orthogonal polynomials. More recently, Akaike’s results were discussed in detail and complemented by bounds on the “oscillations” that can occur in every step [6] (see also [5]), and another proof of the Forsythe-Motzkin conjecture was given in the context of the steepest descent method for matrix functions [1].

Forsythe conjectured that the asymptotic behavior of the optimum $s$-gradient method is similar for all $s$ in the sense that the sets of limit points of the normalized errors with odd and even indices each consist of a single point. Finding a proof of his conjecture is of great interest since the conjectured result would lead to a better understanding of the asymptotic behavior of gradient descent methods for minimizing positive definite quadratic functions, and possibly any sufficiently smooth function which is locally like Forsythe’s model function $f$.

After formulating the Forsythe conjecture in modern notation, we will discuss the state-of-the-art of the attempts to prove it, including particularly the cases $s = 1$ (see above) and $s = 2$ (see [8]). Moreover, we will show that some of the theorems obtained by Forsythe are not restricted to the optimum $s$-gradient method and positive definite functionals, so that the conjecture can in fact be generalized to certain variants of Krylov subspace methods for nonsymmetric linear algebraic systems. We will present new results concerning the “cross iterations” of the restarted GMRES method that was described in [3], and we will extend these to the restarted Arnoldi method. Results obtained in this context can give further insight into the worst-case convergence behavior of the GMRES and Arnoldi methods, and into the properties of the initial vectors for which this behavior is attained, which is also a notoriously difficult and largely open problem.
References


Quantum Linear System Solver

Dong An, Lin Lin

Department of Mathematics, University of California, Berkeley
Computational Research Division, Lawrence Berkeley National Laboratory
https://math.berkeley.edu/~linlin/

Abstract

With the emergence of near-term quantum computing architectures, such as those being developed in IBM, Google, Rigetti etc, currently there is rapidly increasing interest in designing efficient algorithms which are suitable for quantum computers. Quantum computers are devices based on quantum physical systems, and they can perform computations in a very different way from classical computers. Several classes of operations can be performed much more efficiently on quantum computers than on classical computers, and vice versa. For example, on quantum computers, computing a generic matrix-vector multiplication is very hard and copying an unknown state is explicitly forbidden (called the no-cloning theorem), but it is extremely efficient to perform certain unitary transformations. This different architecture makes it possible for quantum algorithms to achieve speedups over classical algorithms for certain problems, such as factoring and unstructured search.

Quantum algorithms can also potentially significantly outperform classical algorithms in numerical linear algebra. Take the linear system solver for example, for a linear system $Ax = b$, where $A \in C^{N \times N}$ and $b \in C^N$, the cost of any generic algorithm on a classical computer scales at least as $O(N)$. However, certain quantum algorithms promise to solve the linear with $O(\text{poly log}(N))$ cost, i.e. before a classical computer even finishes loading the vector $b$ into the memory!

In the past few years, there has been significant progresses on solving the quantum linear system problem (QLSP). The goal of QLSP is to efficiently compute the normalized solution $|x\rangle = A^{-1} |b\rangle / \|A^{-1} |b\rangle \|_2$ of the linear system, or the expectation value $\langle x |O| x\rangle$ of some operator, where $A, O \in C^{N \times N}$, $|b\rangle \in C^N$ is a normalized vector, and $\langle x |$ means the conjugate transpose of $|x\rangle$. The groundbreaking Harrow, Hassidim, and Lloyd algorithm obtains $|x\rangle$ with cost $O(\text{poly(log}(N)) \kappa^2 / \epsilon)$. Here $\kappa$ is the condition number, $\epsilon$ is the target accuracy. Recent progresses based on linear combination of unitaries and quantum signal processing have further improved the scaling to $\tilde{O}(d \kappa^2 \text{poly}(\text{log}(d \kappa / \epsilon)))$, assuming the matrix is $d$-sparse. All of these quantum algorithms achieve significant speedups over classical algorithms with respect to the dimension $N$.

Adiabatic quantum computing (AQC), which is an interesting method weaving together eigenvalue problems, differential equations and linear systems, can offer an alternative route for solving QLSP. Adiabatic quantum computing is designed to approximate an eigenstate of a Hermitian matrix $H_1$ corresponding to eigenvalue $\lambda_1$ by propagating the adiabatic evolution

$$\frac{1}{T} \partial_s |\psi_T(s)\rangle = H(s) |\psi_T(s)\rangle$$

where $0 \leq s \leq 1$, $H(s)$ is Hermitian for any $s$ with $H(1) = H_1$, and the parameter $T > 0$ is called the runtime of AQC. Adiabatic theorem demonstrates that if there exists a path of the eigenvalue of $H(s)$, namely $\lambda(s)$, such that $\lambda(1) = \lambda_1$ and $\lambda(s)$ is separated from the rest of the spectrum by a uniformly positive gap $\Delta(s) \geq \Delta_0 > 0$, then starting from the eigenstate of $H(0)$ corresponding to $\lambda_0 = \lambda(0)$, the final solution $|\psi_T(1)\rangle$ will approximate the desired eigenstate when the runtime $T$ goes to infinity. To formulate QLSP into AQC setup, [2] provides a construction
\[ H(s) = (1 - s)H_0 + sH_1 \] where one of the eigenstate of \( H_1 \) contains the information of the QLSP solution, and the connected eigenstate of \( H_0 \) is easy to prepare. The special case of such construction is when \( A \) is a Hermitian positive definite matrix, \( H_0 = [0; Q_b; Q_b, 0] \) and \( H_1 = [0; Q_b A; A Q_b, 0] \) with \( Q_b = I - |b\rangle \langle b| \), then 0 is an eigenvalue for all \( H(s) \), separated from the rest of the spectrum in a proper-defined subspace by \( \Delta(s) \geq 1/\kappa \). \([0; b \rangle \) and \([0; x \rangle \) are the corresponding connected eigenstate of \( H_0 \) and \( H_1 \), respectively. Therefore, starting from \([0; b] \), AQC can be used to approximate the solution of QLSP.

AQC can be efficiently implemented on a quantum computer since the adiabatic evolution is unitary. Specifically, it can be performed via efficient Hamiltonian simulation, the cost of which can be bounded by \( \tilde{O}(d T \log(d T / \epsilon)) \). Furthermore, the quantum adiabatic theorem [3] provides an upper bound of \( \epsilon \) as

\[
\begin{align*}
C \{ \frac{\|H^{(1)}(0)\|_2}{T \Delta^2(0)} + \frac{\|H^{(1)}(1)\|_2}{T \Delta^2(1)} + \frac{1}{T} & \int_0^1 \left( \frac{\|H^{(2)}(s')\|_2}{\Delta^2(s')} + \frac{\|H^{(1)}(s')\|_2}{\Delta^2(s')} \right) ds' \} \\tag{2}
\end{align*}
\]

therefore the worst case bound as \( T = \mathcal{O}(\kappa^3 / \epsilon) \), and the total computational cost becomes \( \tilde{O}(d \kappa^3 / \epsilon) \), achieving speedup with respect to the dimension but extremely sensitive to the condition number.

We find that the efficiency of AQC can be improved by carefully rescheduling the adiabatic evolution. The key observations are that the accuracy of AQC depends not only on the gap \( \Delta(s) \) but also on the derivatives of \( H(s) \), as revealed in the estimate (2), and that the gap \( \Delta(s) \) only touches the lower bound \( 1/\kappa \) at the final time while remaining relatively large for most time of the propagation. Therefore we construct \( H(s) = (1 - f(s))H_0 + f(s)H_1 \) for a specific class of smooth monotonic functions \( f : [0, 1] \to [0, 1] \), allowing the Hamiltonian \( H(s) \) to slow down when the gap is close to 0, then the error bound of AQC is improved. We prove that the runtime to achieve desired accuracy \( \epsilon \) is only \( T = \mathcal{O}(\kappa / \epsilon) \), and the total computational cost becomes \( \tilde{O}(d \kappa / \epsilon) \), which achieves speedup with respect to the dimension over classical algorithms, together with the optimal scaling with respect to the condition number to be expected on quantum computers. To our knowledge, this is the first algorithm yielding \( \tilde{O}(\kappa / \epsilon) \) cost in the literature, and it does not involve the complicated variable-time amplitude amplification (VTAA) procedure to reduce the dependence on the condition number \( \kappa \).

Due to the natural connection between eigenvalue problems and optimization problems, it is possible to solve QLSP with variational algorithms, one of which is the quantum approximate optimization algorithm (QAOA). QAOA is a generalization of AQC, in the sense that the dynamics of AQC can be approximately represented by QAOA using an operator splitting scheme. This implies the quantum part of computational cost is at most \( \tilde{O}(\kappa / \epsilon) \). This is a very promising method to solve QLSP, as the numerically observed scaling is only \( \tilde{O}(\kappa \text{poly}(1/\epsilon)) \). Since quantum linear solvers cannot perform generic linear systems with scaling better than \( \tilde{O}(\kappa) \), this will likely be the best achievable asymptotic scaling, up to logarithmic factors.

References


Polynomial Preconditioning with the GMRES Polynomial

Jennifer Loe, Ron Morgan, Mark Embree, Heidi Thornquist, Erik Boman
Mathematics, Baylor University
https://blogs.baylor.edu/jennifer_loe/

Abstract

We present a new polynomial preconditioner that is simple to compute and effective for general linear systems $Ax = b$. The polynomial is based upon the GMRES polynomial and is applied using harmonic Ritz values as roots. For high degrees, the polynomial can be made stable by adding extra copies of roots. When applied to GMRES, the polynomial preconditioner can greatly improve the solve time for convergence. In particular, it reduces the proportion of time spent performing dot products, which require expensive global communication and synchronization in parallel computing. The polynomial can even be used to accelerate existing preconditioners such as ILU.

Polynomial preconditioning has frequently been studied (see, for example, [1, 2]) but is rarely used in practice. This may be due to stability concerns [3] or difficulty obtaining a good polynomial, especially for nonsymmetric matrices where estimates of the spectrum of $A$ are required. Our polynomial preconditioner $p(A)$ is generated by running an initial cycle of GMRES and computing the harmonic Ritz values $\theta_i$ from the resulting upper-Hessenberg matrix. Then the preconditioned operator of degree $d$ is implemented as

$$Ap(A) = I - \prod_{i=1}^{d} \left( I - \frac{A}{\theta_i} \right).$$

(1)

We give a related formula for applying $p(A)$ alone to obtain the solution vector at the end of the iteration. This implementation is generally stable for moderate degrees, but sometimes it can become ill-conditioned. To remedy this, for each harmonic Ritz value $\theta_j$, we monitor the value $pof(j) \equiv \prod_{i=1, i \neq j}^{d} \left| 1 - \frac{\theta_j}{\theta_i} \right|$, which is related to the derivative of $ap(\alpha)$ at $\theta_j$. The value $pof(j)$ determines when to append an extra copy of the factor $(I - A/\theta_j)$ to (1). We give an algorithm to automate root-adding. These added roots can make high-degree polynomials to be very stable. For example, let $A$ be the matrix Goodwin_071 from the SuiteSparse Matrix collection [4] and $b$ a random vector. A degree 80 polynomial applied to GMRES is initially unstable; the true residual stalls out at $3.8 \times 10^{-3}$, while the short residual converges. With 4 added roots, the true residual converges to $1 \times 10^{-8}$ in 36 iterations.

We demonstrate that the polynomial can effectively precondition several practical problems when applied to GMRES and that it can significantly decrease dot products that are expensive for parallel computations. For the following example, we use an implementation of the polynomial in the software Trilinos and run GMRES(100) with two passes of classical Gram-Schmidt orthogonalization per iteration. For the matrix ML_Geer from the Janna collection on SuiteSparse, GMRES without preconditioning does not converge within 100,000 iterations. With a polynomial of degree 20, the problem converges in 12,900 iterations and 3214 seconds over 32 MPI processes. Increasing the polynomial degree to 80 gives substantial improvement. The problem now converges in 200 iterations, 16,970 matrix-vector products, and 197 seconds. This is almost sixty-five times reduction in dot products over the degree 20 polynomial. We discuss the relationship of polynomial preconditioning to communication-avoiding GMRES variants and propose directions for further investigation.
The polynomial can also be composed with standard preconditioners. For the matrix Transport, a degree 60 polynomial reduces solve time for GMRES(100) from 1042 seconds over 32 MPI processes to 14.51 seconds. With an ILU(1) preconditioner alone, the solve time is 55.9 seconds, but composing the ILU preconditioner with a degree 20 polynomial reduces the solve time to 10.22 seconds. Thus, polynomial preconditioning may be especially useful for problems where an existing preconditioner needs an extra push.

We also will compare polynomial preconditioned GMRES to FGMRES and BiCGStab. Ultimately, we assert that the GMRES polynomial is a robust and reliable preconditioner that is versatile for many practical problems.

References


Subspace Acceleration for Convex Optimization Over the Numerical Range

Daniel Kressner, Ding Lu, and Bart Vandereycken

Department of Computer Science, University of California, Davis

Abstract

We consider the optimization problem in which a continuous convex function $F : \mathbb{R}^2 \to \mathbb{R}$ is to be minimized over the joint numerical range of two Hermitian matrices $A$ and $B$:

$$\min_{y \in W(A,B)} F(y) \quad \text{and} \quad W(A, B) = \left\{ (x^H Ax, x^H Bx) : x \in \mathbb{C}^n, \|x\|_2 = 1 \right\} \subset \mathbb{R}^2.$$

Some particular examples of such problems include the Crawford number computation and max ratio minimization problems.

Due to the convexity in $W(A, B)$, the minimizer can be solved by convex optimization, which usually involves repeated computation of Hermitian eigenvalue problems. When the matrices are of large size, such methods will become prohibitively expensive, hence, necessitating the use of subspace acceleration techniques as proposed in this talk.

The basic idea of subspace technique is to reduce the matrix dimension by projection: in the definition of $W(A, B)$, by restricting $x \in U$ to a $k$-dimension subspace $U$, we obtain

$$W(A_k, B_k) = \left\{ (z^H A_k z, z^H B_k z) : z \in \mathbb{C}^k, \|z\|_2 = 1 \right\} \subset W(A, B),$$

where $A_k = U_H A U$, $B_k = U_H B U$, and $U \in \mathbb{C}^{n \times k}$ is an orthogonal basis of $U$. Typically $k \ll n$, so optimizing $F(y)$ over $W(A_k, B_k)$ can be done efficiently. The approximate minimizer obtained can then be improved through iteratively updating the projection subspaces.

In this talk, we consider two particular schemes for subspace updating. The first one applies greedy eigenvector-sampling, which requires to solve a Hermitian eigenvalue problem at each iteration. This scheme has proven superlinear convergence with an order of convergence $1 + \sqrt{2} \approx 2.4$. The second one is based on local gradient descending, which requires only a few matrix-vector multiplications at each iteration. In addition, it allows for block implementation and preconditioning for further acceleration. The performance of both algorithms will be demonstrated by numerical examples, with applications in computing coercivity constants of boundary integral operators and solving multicast beamforming problems.
Computing the Eigenvalues of Large Hadamard Matrix Functions using Low Rank Approximations

Omar De La Cruz Cabrera, Thomas Mach, and Lothar Reichel

Kent State University
https://sites.google.com/site/thomasmach/

Abstract

Let \( \{u_i\}_{i \in I} \) be a set of points in \( \mathbb{R}^n \), say 50,000–500,000 points. We will analyze the relationship between these points by computing the dominant eigenvalues and corresponding eigenvectors of \( M_\rho \in \mathbb{R}^{|I| \times |I|} \) with

\[
M_\rho|_{ij} = \exp(-\rho \|u_i - u_j\|^2),
\]

where \( \rho \) is a positive real parameter. We are interested in the eigenvalues for an interval \([\rho_a, \rho_b]\).

The matrix \( M_\rho \) is a Hadamard matrix function [3, Sect. 6.3] (or entry-wise matrix function—think of Matlab’s exp instead of expm) of the matrix \( D \), with

\[
D_{ij} = \|u_i - u_j\|^2.
\]

That is if \( f \) is an analytic function, like \( \exp(x) \), with \( f_{ij}(x) = \sum_{\ell=0}^\infty A_{\ell,ij} x^\ell \), then

\[
f_H(M) = \sum_{\ell=0}^\infty A_{\ell} \circ M^\ell, \quad \text{with} \quad M^\ell = \underbrace{M \circ M \circ \cdots \circ M}_{\ell \text{ terms}}
\]

is a Hadamard matrix function, where \( A \circ B \) is the Hadamard product of \( A \) and \( B \), the entry-wise product \( A \circ B|_{ij} = A_{ij} B_{ij} \). The computation of the eigenvalues of \( M_\rho \) is significantly more difficult than for standard matrix functions defined through an eigenvalue decomposition, \( f(M) = V f(\Lambda) V^{-1} \) with \( A = VAV^{-1} \) the eigenvalue decomposition of \( M \). A similar statement providing us the eigenvalues for all \( \rho \) does not exist for Hadamard matrix function.

To obtain information about the eigenvalues of \( M_\rho \), we will discretize the interval \([\rho_a, \rho_b]\) and compute the eigenvalues for all \( \rho_i \) separately. This decision is due to the size of the matrix that does not allow us to use an approach similar to the analytic singular value decomposition [2].

Therefore, it is important that the solution of each matrix eigenvalue problem is as efficient as possible. It has been shown that \( M_\rho \) is symmetric positive definite [5]. Thus we use an incomplete Cholesky decomposition to find a low-rank approximation of \( M_\rho \). To control the accuracy of the low-rank approximation we use diagonally pivoting. This procedure can be regarded as the adaption of adaptive cross approximation [1, 6] to symmetric positive definite matrices. Hence we also have a similar behavior, especially only \( \mathcal{O}(|I|) \) entries of \( M_\rho \) have to be computed to obtain a rank \( k \) approximation \( LL^T \).

We then use a singular value decomposition of \( L \), which can be computed in \( \mathcal{O}(|I|k^2) \). This reveals the dominant eigenvalues of \( M_\rho \). An error analysis demonstrates that the errors of the eigenvalue approximations are small and can be controlled by choosing \( k \) adaptively based on the size of the next diagonal pivot. The overall complexity depends only linearly on the number of points.

We track the eigenvalues from \( \rho_i \) to \( \rho_{i+1} \) based on the angle of the corresponding eigenvectors. We match two eigenvalues if the angle between the eigenvectors is small.\(^1\) If we cannot find a matching

\[\text{The eigenvectors to different eigenvalues of a symmetric matrix are orthogonal.}\]
partner for one of the dominant eigenvalues, then we refine the discretization by adding a point in the middle between $\varrho_i$ and $\varrho_{i+1}$. The tracking is based on the observation that the entries of $M_\varrho$ are analytic functions in $\varrho$ and thus the eigenvalues and eigenvectors are analytic functions, see Kato [4, pp. 120–122]. Furthermore, symmetric positive definite matrices cannot have defective eigenvalues, which excludes the most difficult cases.

Finally, we will demonstrate the usage of our algorithm on some examples analyzing sets of points. In plain words, the parameter $\varrho$ decides how far we consider points being neighbors. For instance, in a coil two points are near each other if they are neighboring on the wire. However, if we widen our view points on the next winding are nearby as well. We will explain how this behavior can be observed in the eigenvalues of $M_\varrho$.

References


A Natural Vector Space Setting for Block Minimal Basis Pencils

D. Steven Mackey, Javier Pérez, and Philip Saltenberger

Dept. of Mathematics, Western Michigan University
http://homepages.wmich.edu/~dmackey/index.html

Abstract

Over the last several decades, the eigenproblem for matrix polynomials has received a great deal of research attention. One of the main approaches to this problem has been through the notion of linearization, i.e., the conversion of a matrix polynomial into a matrix pencil with exactly the same spectral data, so that more tractable computations can then be done on the pencil. Although there are infinitely many ways to do this spectra-preserving conversion, in the 1900’s attention was focused primarily on just a handful of examples, dominated by what are now known as the Frobenius companion pencils. However, these pencils do not preserve important algebraic structures such as symmetry or palindromicity. Also, with only a few examples to work with, the possibility of trying to optimize the numerical properties of the chosen linearization could not even be sensibly formulated. Consequently, there was strong motivation to find simple ways to significantly broaden the available palette of easily specifiable linearizations.

Two new approaches to the construction of linearizations were thus introduced in the early 2000’s to address these drawbacks of the companion pencils. The first approach led to the Fiedler pencils [10, 1, 5], the second to the ansatz spaces [15, 16]. Since their initial introduction, both of these approaches have been intensively investigated and extended in many additional papers; [2, 3, 12, 13, 17, 21] is a small sampling of these papers. These two families generalize the Frobenius companion pencils in qualitatively different ways; the Fiedler pencils comprise a finite discrete set of linearizations, whereas each ansatz space provides a continuum (vector space) of pencils (sometimes of rather large dimension), almost all of which are linearizations.

More recently, a third approach for generating linearizations has been introduced in [7]; these new linearizations are the block minimal basis pencils (BMB-pencils), which include the special sub-class of block Kronecker pencils. In the relatively short time since their introduction, there has been a very rapid development of the properties and scope of applicability of these BMB-pencils [8, 18, 19, 20], due both to the flexibility of their definition, and to their amenability to analysis of their numerical properties. As their name might suggest, the definition and properties of BMB-pencils are based on the notion of a minimal basis of a rational vector space [11, 14], but more especially on that of dual minimal bases [11, 6]. It has been shown that Fiedler pencils are essentially (up to block permutation) just special cases of block Kronecker pencils, leading to a simplified analysis of the properties of Fiedler pencils [2].

In many papers, block Kronecker pencils are viewed as isolated pencils, one at a time. Or perhaps small perturbations of such pencils into nearby BMB-pencils are considered. The recent paper [9], though, considers block Kronecker pencils collectively, as continuous families with vector space structures. The purpose of this talk, then, is to describe how these ideas can be extended to encompass all BMB-pencils, not just the block Kronecker ones. That is, we show how BMB-pencils can be naturally embedded in ansatz-like vector spaces of pencils (almost all of which are linearizations), thus effecting a unification of all three of the main approaches to generating linearizations of matrix polynomials.
References


Fast sampling of parameterised Gaussian random fields via low-rank approximation

D. Kressner, J. Latz, S. Massei, and E. Ullmann

École Polytechnique Fédérale de Lausanne
www.researchgate.net/profile/Stefano_Massei2

Abstract

A widely used method for drawing a sample from a Gaussian distribution $K := \mathcal{N}(\mathbf{m}, C)$ over $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$, relies on computing $R\mathbf{x} + \mathbf{m}$, where $R$ is the Cholesky factor of the covariance matrix $C$ and $\mathbf{x}$ is a vector drawn from a standard normal distribution.

This work is concerned with the sampling from parameter dependent Gaussian measures on $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n))$ of the form

$$K(\cdot|\theta) := \mathcal{N}(\mathbf{m}(\theta), C(\theta)), \quad \theta \in \Theta,$$

where $\Theta$ is a (possibly high dimensional) parameter space that is associated with some $\sigma$-algebra $\mathcal{A}$ and $\mathbf{m}(\theta), C(\theta)$ are measurable functions taking values on $\mathbb{R}^n$ and on the set of $n \times n$ symmetric positive semidefinite (SPSD) matrices, respectively. Drawing a sample from $K(\cdot|\theta)$ is a two-step mechanism: first a value of $\theta$ is generated from a distribution (e.g. uniform) on $\Theta$. Then, a sample from $\mathcal{N}(\mathbf{m}(\theta), C(\theta))$ is drawn. This paradigm finds application in Bayesian inversion for elliptic partial differential equations with uncertain diffusion coefficients [3].

The central topic of this talk is the efficient approximation of $R(\theta)$ such that $C(\theta) = R(\theta)^T R(\theta)$ $\forall \theta \in \Theta$ in order to speed up the second step of the sampling procedure. When $n$ is large, this is a challenging computational task already in the case of a fixed parameter $\theta$. Indeed, the nonlocality of the covariance operator prevents $C(\theta)$ and consequently $R(\theta)$ from being sparse. On the other hand, in many practical cases, e.g. Matérn kernel covariance operators, $C(\theta)$ exhibits rank structures, independently of the parameter choice.

We propose a new technique for approximating $R(\theta)$ in the case that $C(\theta)$ has a finite separable expansion of the form:

$$C(\theta) = \varphi_1(\theta)C_1 + \cdots + \varphi_s(\theta)C_s, \quad \varphi_j : \Theta \to \mathbb{R}, \quad C_j \in \mathbb{R}^{n \times n}, \quad j = 1, \ldots, s. \quad (1)$$

The core idea is to perform a low-rank approximation of $C(\theta)$ by extending the adaptive cross approximation (ACA) algorithm for an SPSD matrix [2] to a parameter dependent family of SPSD matrices. Our method preserves some of the attracting features of ACA. In particular, it is matrix free in the sense that it requires only a partial evaluation of the entries of $C(\theta)$, for a few values of $\theta$, and avoids forming any $n \times n$ matrix explicitly. The outcome of the procedure is an index set $I \subseteq \{1, \ldots, n\}$ such that $k = |I| \ll n$ and the matrix

$$C_I(\theta) := C(\theta)(:, I)[C(\theta)(I, I)]^{-1}C(\theta)(I,:)$$

verifies $\|C(\theta) - C_I(\theta)\|_F < \epsilon$ for all $\theta \in \Theta$ and a prescribed tolerance $\epsilon$. From the probabilistic point of view, we show that the quantity $\epsilon$ is an upper bound for the Wasserstein distance between the families of measures $\mathcal{N}(\mathbf{m}(\theta), C(\theta))$ and $\mathcal{N}(\mathbf{m}(\theta), C_I(\theta))$. Finally, an approximation of $R(\theta)$ is given by $\tilde{R}(\theta) := C(\theta)(:, I)R_I(\theta)^{-1}$ where $R_I(\theta)$ is the Cholesky factor of the $k \times k$ matrix $C(\theta)(I, I)$.

We suggest a sampling strategy that operates in a model order reduction fashion. In the offline phase we compute the index set $I$ by means of the algorithm previously described. In the online phase,
for every sample of $\theta$ we build the (approximate) Cholesky factor $\hat{R}(\theta)$ of $C(\theta)$, the vector $m(\theta)$ and we compute $\hat{R}(\theta)x + m(\theta)$ for a standard normal random vector $x$. The overall computational cost depends only linearly on the size $n$ of the covariance matrices.

We also address the case that $C(\theta)$ does not admit a finite separable expansion with respect to the parameter $\theta$. First, we test the use of the empirical interpolation method [1] for retrieving an approximate expansion of the form (1). Then, we analyze the robustness of the proposed algorithm when applied to an inexact separable expansion of $C(\theta)$.

The performances of the sampling strategy are numerically tested on several choices of covariance kernel operators and demonstrate a significant gain with respect to state of the art techniques.

References


Computing Eigenvectors in Floating Point Arithmetic

Teresa Laudadio, Nicola Mastronardi, and Paul Van Dooren

Istituto per le Applicazioni del Calcolo “M. Picone”, Consiglio Nazionale delle Ricerche
https://users.ba.cnr.it/iac/irmanm21/index.htm

Abstract

The $QR$ method is the customary algorithm to compute the eigenvalue decomposition of a matrix $A \in \mathbb{C}^{n \times n}$ of moderate size. To reduce the complexity of the algorithm, the matrix $A$ is first reduced to an Hessenberg matrix $H$ by means of a sequence of $n - 1$ Householder similarity transformations. The second step is to perform a series of $QR$ sweeps with so-called shifts.

Different choices of shifts have been considered in the literature [2].

If such a shift is an exact eigenvalue $\lambda_0$ of $A$, then, in infinite precision arithmetic, this eigenvalue will appear on the diagonal of the transformed Hessenberg matrix $\tilde{H}$ and will be decoupled from the remaining part of the Hessenberg matrix, thus resulting in a deflation. But it is well known that in floating point arithmetic the so-called perfect shift can get blurred [6] and that the eigenvalue $\lambda_0$ can then not be deflated.

In this work, we consider two strategies for modifying a $QR$ sweep so that the eigenvector associated to $\lambda_0$ is computed with higher accuracy and the deflation is successful in most cases.

The first approach is based on the preliminary computation of the eigenvector $x$ corresponding to $\lambda_0$, such that the residual $(H - \lambda_0 I)x$ is sufficiently small.

The eigenvector is then transformed to a unit vector $e_1$ by a sequence of Givens transformations, which are also performed on the Hessenberg matrix. Notice that what we just described is a “backward” $QR$ step where we transform the eigenvalue $\lambda_0$ to position $(1,1)$ in $\tilde{H}$, whereas the standard (forward) $QR$ step moves it to position $(n,n)$.

This approach has been applied to compute the Generalized Null-space of a matrix [3].

The second approach applies to tridiagonal matrices. It is based on two different ways to compute the eigenvector $x_0$ associated to $\lambda_0$. The first way computes the eigenvector performing one sweep of the forward $QR$ method, whereas the second one makes use of the backward $QL$ method, a variant of the $QL$ method applied from the bottom–right corner of the tridiagonal matrix to the upper–left one.

Unfortunately, both methods can suffer from forward instability when working in floating point arithmetic. We analyzed the reason of such a behaviour and developed an algorithm, based on a suitable combination of the aforementioned methods, to overcome this phenomenon.

The latter approach has been successfully applied to a variety of non–symmetric tridiagonal matrices. In particular, it was applied to compute the matrix eigenvector corresponding to the eigenvalues of tridiagonal matrices associated to Bessel polynomials [1].

References


Multigrid Strategies for both Isotropic and Anisotropic Fractional Diffusion Problems

Mehdi Dehghan, Marco Donatelli, Rolf Krause, Mariarosa Mazza, Hamid Moghaderi, Ken Trotti

Department of Science and High Technology, University of Insubria

https://sites.google.com/site/mariarosamazza1/

Abstract

The mathematical theory of fractional calculus roots back to the 18th century, but it was only since the 80s that the fractional paradigm started to be largely employed in the applications for modeling a broad class of systems and processes arising in physics, control theory, as well as in engineering, and life sciences.

A kind of fractional models that is establishing itself relentlessly are the Fractional Diffusion Equations (FDEs) which enable the modeling of anomalous diffusion phenomena appearing in several applicative fields. As an example, we briefly describe how FDEs apply to diffusion weighted MRI. Diffusion weighted MRI allows the mapping of the diffusion process of water molecules in biological tissues, and is used clinically to detect and characterize neurodegenerative, malignant and ischemic diseases. Since molecular diffusion in tissues is affected by the interactions with many obstacles that are highly heterogeneous, standard diffusion models typically provide a limited description. On the contrary, recent results indicate that the use of (possibly anisotropic) fractional models is promising for classifying anomalous diffusion in tissues with developing pathology [6].

Compared with standard diffusion models, FDEs provide then an improved physical description of the considered phenomenon. On the other hand, this translates in a harder numerical treatment of the corresponding discretized problems. Indeed, even when standard local discretization methods like finite differences or finite elements are adopted, the non locality of the fractional operators causes absence of sparsity in the discretization matrices. This makes FDEs computationally more demanding than standard diffusion equations and stresses the need of proper numerical linear algebra tools.

In this regard, a large part of the literature has been focused on the structure of the discretization matrices and on how to exploit it when dealing with the solution of the associated linear systems [2, 5, 7]. Our work fits within this framework. Precisely, we are interested in a second order accurate finite difference discretization [11] of the following two-dimensional initial-boundary value time-dependent space-FDE problem

\[
\begin{cases}
\frac{\partial u(x, y, t)}{\partial t} = d_+ (x, y, t) \frac{\partial^\alpha u(x, y, t)}{\partial x^\alpha} + d_- (x, y, t) \frac{\partial^\alpha u(x, y, t)}{\partial x^{-\alpha}} + e_+ (x, y, t) \frac{\partial^\beta u(x, y, t)}{\partial y^\beta} + e_- (x, y, t) \frac{\partial^\beta u(x, y, t)}{\partial y^{-\beta}} + v(x, y, t), & (x, y, t) \in \Omega \times [0, T], \\
u(x, y, t) = 0, & (x, y, t) \in (\mathbb{R}^2 \setminus \Omega) \times [0, T], \\
u(x, y, 0) = u_0(x, y), & (x, y) \in \Omega,
\end{cases}
\]

where \( \Omega = (a_1, b_1) \times (a_2, b_2) \) is the space domain, \([0, T]\) is the time domain, \( d_\pm, e_\pm \geq 0 \) are the diffusion coefficients, \( v \) is the forcing term, and \( \frac{\partial^\alpha u}{\partial x^\alpha}, \frac{\partial^\beta u}{\partial y^\beta} \) are the fractional derivatives, with \( \alpha, \beta \in (1, 2) \) the fractional derivative orders in \( x- \) and \( y- \)variable, respectively.

Our numerical proposal is driven by the following two facts: (i) independently of \( \alpha, \beta \) and in presence of uniform grids, the discretization matrices corresponding to (1) have a block-Toeplitz-Toeplitz-blocks-like structure [5]; (ii) according to the choice of \( \alpha, \beta \), problem (1) can show an intrinsic anisotropy along the coordinates.
Item (i), together with the well-known negative results on multilevel circulant preconditioning [10], guides our choice towards a multigrid approach. Item (ii), in turn, asks for a careful selection of both projector and smoother in accordance with the isotropic/anisotropic character of the problem.

In the following we give a precise account of what is our proposal in both isotropic and anisotropic scenarios.

- **Isotropic case.** Based on the spectral analysis of the coefficient matrices [4], we define a multigrid method with classical linear interpolation as grid transfer operator and damped-Jacobi as smoother, and we show that the corresponding two-grid method converges linearly [8]. That same spectral analysis is employed also for estimating the parameter of the damped-Jacobi in such a way that it satisfies the smoothing property.

Starting from these results we build two different multigrid preconditioners that are remarkably effective when the fractional orders do not differ too much from each other (isotropic case). In particular, one of these is a two-dimensional scaled-Laplacian preconditioner that reveals computationally very attractive when the fractional orders are both close to 2.

- **Anisotropic case.** The above approach shows two main drawbacks: first, the theoretical estimation of the relaxation parameter is available only for constant diffusion coefficients; second, the multigrid theory does not apply anymore for anisotropic problems.

To overcome these drawbacks, inspired by the ideas in [3, 9] on multigrid methods for integer order anisotropic diffusion equations, in [1] we build a multigrid (block-banded-banded-block) preconditioner that has relaxed Jacobi as smoother and whose grid transfer operator is obtained with a semi-coarsening technique [3]. The Jacobi relaxation parameter is now estimated by using an automatic spectral-based procedure that extends also to the variable coefficients case. A further improvement in the robustness of the proposed method with respect to the anisotropy of the problem is attained employing the V-cycle with semi-coarsening as smoother inside an outer full-coarsening [9].

**References**


Robust representations of passive systems in port-Hamiltonian form

Volker Mehrmann and Paul Van Dooren

Inst. f. Math., TU Berlin
www.math.TU-Berlin.de/?id=76888

Abstract

It is well known that under some weak conditions every stable and passive linear continuous-time dynamical system of the form \( \dot{x} = Ax + Bu, \ y = Cx + Du \), with equal input and output dimension, can be transformed via a state-space transformation to a port-Hamiltonian system of the form \( \dot{\tilde{x}} = (J - R)Q\tilde{x} + (F - P)u \), \( y = (F + P)^TQ\tilde{x} + Du \), where \( J = -J^T \), \( Q = Q^T \) positive definite and the passivity matrix

\[
W = W^T = \begin{bmatrix}
R & P \\
P^T & \frac{1}{2}(D + D^T)
\end{bmatrix}
\]

positive is semidefinite, see [1]. However, this representation is not unique, and thus the freedom in the transformation, which rests in the solution set of a linear matrix inequality, can be used to determine an optimally robust representation. We determine such optimal representations for the discrete-time case (see [3]) and for the continuous-time case (see [2]) by optimizing quantities like the distance to non-passivity, or instability.

The presented techniques can also be used to find the nearest passive system to a given non-passive one.

References


The infinite Lanczos method for symmetric nonlinear eigenvalue problems

Giampaolo Mele
Department of Mathematics, KTH Royal Institute of Technology in Stockholm
https://people.kth.se/~gmele/

Abstract

We consider the symmetric nonlinear eigenvalue problem (NEP) which consists of computing 
\((\lambda, x) \in D \times \mathbb{C}^n \setminus \{0\}\) such that

\[ M(\lambda)x = 0, \]

with \(D \subset \mathbb{C}\) open disk centered in the origin and \(M : D \to \mathbb{C}^{n \times n}\) analytic that fulfills \(M(\lambda)^T = M(\lambda)\) for all \(\lambda \in D\). The vector \(x\) is called eigenvector and the scalar \(\lambda\) eigenvalue. The targeted eigenvalues are located in \(D\).

It is shown in [2] that the NEP is equivalent to the following linear infinite dimensional eigenvalue problem:

\[
\begin{pmatrix}
-M_0 \\
I \\
I \\
I \\
\vdots
\end{pmatrix}
\begin{pmatrix}
\frac{\lambda^0}{0!} x \\
\frac{\lambda^1}{1!} x \\
\frac{\lambda^2}{2!} x \\
\frac{\lambda^3}{3!} x \\
\vdots
\end{pmatrix} = \lambda
\begin{pmatrix}
M_1 \\
\frac{1}{2} M_2 \\
\frac{1}{3} M_3 \\
\frac{1}{4} M_4 \\
\vdots
\end{pmatrix}
\begin{pmatrix}
\frac{\lambda^0}{0!} x \\
\frac{\lambda^1}{1!} x \\
\frac{\lambda^2}{2!} x \\
\frac{\lambda^3}{3!} x \\
\vdots
\end{pmatrix}.
\]

(1)

The matrices \(\{M_j\}_{j=0}^{\infty}\) are the coefficients of the Taylor series expansion

\[ M(\lambda) = \sum_{j=0}^{\infty} \lambda^j M_j. \]

The Arnoldi method can be directly applied to the companion pencil (1), the resulting method is called infinite Arnoldi [2]. Unfortunately, this method does not exploit the symmetric structure of the NEP. The main issue concerns the fact that the companion pencil (1) is never symmetric.

Our approach is based on symmetrizing the companion pencil. This can be done as stated in the following theorem, which can be seen as an extension of the previous results [3, 5].

**Theorem** By left-multiplying (1) by the matrix

\[
\begin{pmatrix}
I \\
c_{1,1} M_2 & c_{1,2} M_3 & c_{1,3} M_4 & c_{1,4} M_5 & \ldots \\
c_{2,1} M_3 & c_{2,2} M_4 & c_{2,3} M_5 \\
c_{3,1} M_4 & c_{3,2} M_5 \\
c_{4,1} M_5 \\
\vdots
\end{pmatrix},
\]

\[
c_{i,1} := \frac{1}{i+1} \quad i \geq 1
\]

\[
c_{i-1,j} := \frac{j c_{i,j-1}}{i} \quad i, j > 1
\]
we obtain the symmetric eigenvalue problem

\[
\begin{pmatrix}
-M_0 & c_{1,1}M_2 & c_{1,2}M_3 & c_{1,3}M_4 & \ldots \\
c_{2,1}M_3 & c_{2,2}M_4 & & & \\
c_{3,1}M_4 & & & & \\
& \vdots & & & \\
\end{pmatrix}
\begin{pmatrix}
\lambda_0^0 \\
\lambda_1^1 \\
\lambda_2^2 \\
\vdots \\
\end{pmatrix}
= \lambda
\begin{pmatrix}
\frac{1}{2}M_1 & \frac{1}{2}M_2 & \frac{1}{3}M_3 & \frac{1}{4}M_4 & \ldots \\
\frac{c_{1,1}}{2}M_2 & \frac{c_{1,2}}{3}M_3 & \frac{c_{1,3}}{4}M_4 & & \\
\frac{c_{2,1}}{4}M_4 & & & & \\
\vdots & & & & \\
\end{pmatrix}
\begin{pmatrix}
\lambda_0^0 \\
\lambda_1^1 \\
\lambda_2^2 \\
\vdots \\
\end{pmatrix}
\]  

(2)

Symmetric generalized eigenvalue problems \(Ax = \lambda Bx\), where \(A, B \in \mathbb{C}^{n \times n}\) are symmetric matrices, can be solved with a variation of the Lanczos method, which is referred as to indefinite Lanczos \cite{4}. This corresponds to apply the Arnoldi method to the matrix \(A^{-1}B\) replacing the Euclidean scalar product with the bilinear form \(<x, y>_B:= x^TBy\). Due to the fact that the matrix \(A^{-1}B\) is self adjoint with respect to \(<\cdot, \cdot>_B\), the orthogonal basis of the Krylov space fulfills a three term recurrence. In particular, after \(k\) iterations, the following Arnoldi-like factorization is obtained

\[
A^{-1}BQ_k = Q_{k+1}T_{k+1,k},
\]

\[
Q_{k+1}^T BQ_{k+1} = \Omega_{k+1},
\]

where \(T_{k+1,k}\) is the matrix containing the orthogonalization coefficients and \(\Omega_{k+1}\) is the diagonal matrix with elements \(\omega_j = q_j^TBq_j\) for \(j = 1, \ldots, k + 1\). The Ritz procedure consists of solving the projected problem \(T_kz = \lambda\Omega_kz\) and extract eigenpair approximations as \((\lambda, Q_kz)\).

The indefinite Lanczos method is applicable to the symmetrized companion pencil (2), which is defined by infinite dimensional matrices, without introducing any truncation or approximation. The only restriction is that the starting vector of the Krylov sequence has to have a finite number of nonzero elements. Due the the loss of orthogonality, which is in general present in Lanczos methods, the matrices \((T_k, \Omega_k)\) are affected by high roundoff errors. This results in a slow convergence if the eigenpair are extracted with the Ritz procedure. This issue is resolved by extracting eigenpair approximations with the nonlinear Rayleigh-Ritz procedure, which consists of solving the projected problem \(V^TM(\lambda)Vz = 0\) where \(V\) is an orthogonal matrix whose columns span a space containing a good approximation of the eigenvectors of the original NEP. Due to the structure of the matrices in (2), a natural candidate for the matrix \(V\) is an orthogonal basis of the first block row of \(Q_m\).

The resulting method, referred as to infinite Lanczos, is fully described in \cite{1} and is implemented in the Julia package NEP-PACK \cite{6}, which is a package for NEPs.

References


https://github.com/nep-pack/NonlinearEigenproblems.jl
On prescribing the convergence behaviour of the Conjugate Gradient algorithm

Gérard Meurant

CEA, France (retired)
gerard-meurant.pagesperso-orange.fr/

Abstract

The Conjugate Gradient (CG) algorithm of M.R. Hestenes and E. Stiefel [3] is the algorithm of choice for solving iteratively linear systems \( Ax = b \) with a symmetric positive definite (SPD) matrix. In this paper we are interested in studying different ways of constructing real SPD matrices \( A \) of order \( n \) and real right-hand sides \( b \) for which the CG algorithm has a prescribed residual norm convergence curve as well as prescribed \( A \)-norms of the error. This means that we can construct linear systems with a fast convergence of the residual norm and a slow convergence for the \( A \)-norm of the error or vice-versa. This problem was already considered in [3] but only for the residual norms.

First, we briefly recall what is known so far for the Full Orthogonalization Method (FOM) [4], [5], a Krylov method for nonsymmetric systems which reduces to CG when the matrix is SPD. We rely on results from Greenbaum and Strakoš [2] to construct a class of simple nonsymmetric linear systems with a prescribed FOM residual norm convergence curve. This is different from the construction that was done in [1].

Then, using these results, we consider the symmetric case for which we construct matrices depending on \( n \) parameters. We show that, if some constraints on the parameters are satisfied, these matrices are positive definite. These constructed matrices are inverses of symmetric tridiagonal matrices \( T \), such that CG applied to \( T \) yields prescribed relative residual norms with the right-hand side \( e_1 \), the first column of the identity matrix. Then, a general SPD linear system \( Ax = b \) is obtained from \( A = VTV^T \) and \( b = Ve_1 \) where \( V \) is any orthonormal matrix. There are remaining free parameters that can be chosen to prescribe the convergence curve for the \( A \)-norm of the error.

We also completely characterize the inverses of the tridiagonal matrices produced by the Lanczos algorithm (which is equivalent to CG for SPD matrices) in terms of the CG residual norms and \( A \)-norms of the error. This is of interest since, in the Lanczos algorithm, the iterates are computed as \( x_k = x_0 + \|r_0\|V_{n,k}T_k^{-1}e_1 \) where the columns of \( V_{n,k} \) are the Lanczos basis vectors and \( T_k \) is the principal submatrix of order \( k \) of \( T \). This also allows to obtain expressions and lower bounds for the \( \ell_2 \) norm of the error in CG.

Finally, we show that, contrary to the nonsymmetric case (see [1]), it is not always possible in the symmetric case to prescribe both the residual norms and the eigenvalues of the matrix. The results are illustrated with numerical examples.

References


Computing the Kreiss Constant of a Matrix

Tim Mitchell

Computational Methods in Systems and Control Theory
Max Planck Institute for Dynamics of Complex Technical Systems
http://www.timmitchell.com

Abstract

The Kreiss constant of a square matrix bounds the transient behavior of its associated ordinary differential equation or difference equation. Introduced by Kreiss in 1962 [6], until recently no algorithm has been given to actually compute it, which is equivalent to solving certain global optimization problems. Local optimizers of these problems can be obtained relatively easily via standard optimization techniques, even for large-scale matrices, but the resulting estimates may be arbitrarily bad. In this talk, we present the first algorithms with theoretical guarantees for computing Kreiss constants, using two different approaches to ensure global convergence.

The continuous-time version of the Kreiss Matrix Theorem, after nearly thirty years of refinement by various authors, states that for any matrix $A \in \mathbb{C}^{n \times n}$

$$K(A) \leq \sup_{t \geq 0} \|e^{tA}\| \leq e^{nK(A)},$$

(1)

where the Kreiss constant $K(A)$ has two equivalent definitions

$$K(A) = \sup_{z \in \mathbb{C}, \text{Re} z > 0} (\text{Re} z) \|(zI - A)^{-1}\| \quad (2a)$$

$$= \sup_{\varepsilon > 0} \frac{\alpha_{\varepsilon}(A)}{\varepsilon}, \quad (2b)$$

and the $\varepsilon$-pseudospectral abscissa $\alpha_{\varepsilon}(\cdot)$ is defined by

$$\alpha_{\varepsilon}(A) = \max \{ \text{Re} z : z \in \mathbb{C}, \|(zI - A)^{-1}\| \geq \varepsilon^{-1} \}. \quad (3)$$

For $\varepsilon = 0$, $\alpha_{\varepsilon}(A) = \alpha(A)$, the spectral abscissa of $A$, and clearly $K(A) = \infty$ holds if $\alpha(A) > 0$. If $A$ is normal and $\alpha(A) \leq 0$, then $K(A) = 1$, which is the minimum value $K(A)$ can take.

To date, Kreiss constants have typically been estimated via trial and error, where a user plots (2a) or (2b) to search for global maximizers. In order for this to succeed, one must first find a region that contains a global maximizer and then sufficiently sample this area in order to resolve it. However, since the optimization problems in (2) may have multiple local maximizers and their domains are unbounded, there are no guarantees that a user will even find such a region, let alone adequately sample it in order to resolve a global maximizer. Furthermore, fine-grained sampling will be necessary whenever transient behavior happens on a very fast time scale, making the entire process even more challenging.

Our first approach for computing Kreiss constants [7] is inspired by algorithms for computing the distance to uncontrollability. Given a linear control system

$$\dot{x} = Ax + Bu,$$

(4)

where $A \in \mathbb{C}^{n \times n}$, $B \in \mathbb{C}^{n \times m}$, and both the state $x \in \mathbb{C}^{n}$ and control input $u \in \mathbb{C}^{m}$ are dependent on time, the distance to uncontrollability $\tau(A, B)$ can be computed by solving [2]

$$\tau(A, B) = \min_{z \in \mathbb{C}} \sigma_{\min}( [A - zI \quad B] ).$$

(5)
State-of-the-art methods for computing $\tau(A, B)$ all trace back to a novel 2D level-set test developed by Gu in 2000 [3], which for a given guess $\gamma \geq 0$ for $\tau(A, B)$ and some $\eta \geq 0$, asserts whether there exists any points $z \in \mathbb{C}$ such that

$$
\sigma_{\min} \left( \begin{bmatrix} A - zI & B \end{bmatrix} \right) = \sigma_{\min} \left( \begin{bmatrix} A - (z + \eta)I & B \end{bmatrix} \right) = \gamma.
$$

(6)

If such points exist, then $\gamma \geq \tau(A, B)$ holds and these points can be used to initialize an optimization solver such that it is guaranteed to find a better (lower) local minimizer of (5). This observation led to the optimization-with-restarts algorithm of Burke, Lewis, and Overton [1] for computing $\tau(A, B)$, where minimizers of (5) are found via optimization, and then the level-set test of Gu is used as a globality certificate, to either assert global convergence or provide new starting points for another round of optimization. By noting the similarity of (5) to the inverse of (2a):

$$
K(A)^{-1} = \inf_{z \in \mathbb{C}, \Re z > 0} \sigma_{\min} \left( \frac{zI - A}{\Re z} \right),
$$

(7)

we recently developed an analogous globally-convergent algorithm for computing $K(A)$ [7, §3], which adapts the test of Gu to instead detect pairs of points a distance $\eta$ apart on a given level set of (2a). The main drawback of this approach is that all of these 2D level-set tests are quite expensive, in fact, $O(n^6)$ work, and they can be quite sensitive to rounding errors, which can cause them to fail in practice. While we have additionally shown that the work complexity can be reduced to $O(n^4)$ [7, §3.3], by adapting the asymptotically faster divide-and-conquer technique of [5] for $\tau(A, B)$, this does not appear to be numerically reliable in the context of computing Kreiss constants [7, §6.3].

In our second approach [8], we propose alternative interpolation-based level-set tests in order to ensure global convergence. The core idea here is to construct a new one-variable distance function, which when sufficiently resolved by an interpolant on a finite interval known a priori, certifies whether or not a given point is a global minimizer of (7). If the given point is not a global minimizer, then typically a low-fidelity interpolant will suffice to restart optimization. While this new globality certificate may require evaluating our distance function many times, each evaluation only requires $O(n^3)$ work and can be done in an embarrassingly parallel manner. In practice, these newer level-set certificates can be orders of magnitude faster to compute (even without parallel processing) and seem to be more reliable as well. We have applied these techniques to develop new algorithms for continuous- and discrete-time Kreiss constants as well as the distance to uncontrollability, allowing all of these quantities to be efficiently computed for much larger problems than previously possible. Finally, we note that this new interpolation-based approach can be used to solve other global optimization problems of singular value functions of two real variables, such as $\text{sep}_\lambda$ [4].

References


Deflation for Krylov Solution of Linear Equations

Ron Morgan

Department of Mathematics, Baylor University
https://sites.baylor.edu/ronald_morgan/

Abstract

Krylov methods are popular for large linear equations, but they can struggle for difficult, ill-conditioned matrices. Deflation of eigenvalues is essential in Krylov methods when there either is no preconditioner or the problem is difficult even after preconditioning. An example is in QCD (quantum chromodynamics) where preconditioning is rarely used. Here we review past work on deflation and then mention current developments. This in-progress work is 1) two-grid deflated, restarted BiCGStab/IDR, 2) a deflated multigrid for QCD, and 3) deflation for singular systems.

1 Review of Deflation

For Krylov methods, it is well known that small eigenvalues can slow the convergence. Deflated methods [3] use approximate eigenvectors to remove the effects of these eigenvalues. Many have worked on deflation; references will be given in the talk. There will be also be discussion of multiple right-hand sides [5], changing matrices [7], and highly non-normal matrices (deflation can be explained with pseudo-eigenvectors [6] or with a perturbed GMRES theorem [8]).

Example: We focus on the deflated method GMRES-DR [4], which both computes the needed approximate eigenvectors corresponding to the small eigenvalues and simultaneously uses them to improve solution of the linear equations. A matrix of size \( n = 40,000 \) is generated using finite differences on the unit square with the biharmonic operator \(-u_{xxxx} - u_{yyyy} + 10u_{xxx}\). BiCGStab and IDR do not converge for this ill-conditioned matrix, and GMRES(75) is very slow with 2.4 million matrix-vector products. GMRES-DR(75,25), which generates 25 approximate eigenvectors, converges in 148 thousand. Convergence speeds up once approximate eigenvectors are computed. For a second right-hand side, GMRES-Proj [5] needs only 68 thousand, because it uses the eigenvectors that were found while solving the first system.

2 New Deflation Methods

2.1 Two-grid deflated, restarted BiCGStab/IDR (with Zhao Yang)

The new method uses two grid levels. Approximate eigenvectors are computed with GMRES-DR on the coarse grid, then moved to the fine grid. A deflated BiCGStab (or IDR) is run on the fine grid. This deflation uses a projection over the approximate eigenvectors, but it is not perfect due to the inaccuracy of the eigenvectors and the lack of left eigenvectors. Therefore runs of BiCGStab to a limited tolerance alternate with the projection. For more, including use of deflation for a multigrid preconditioned Helmholtz problem, see https://sites.baylor.edu/ronald_morgan/reports/ (making sure the underscore is copied) and click on “2GridLE”.

Most novel in this approach is the use of a coarse grid to speed up computation of approximate eigenvectors and the restarting of the normally non-restarted methods BiCGStab and IDR.
2.2 Deflation in multigrid for QCD (with Travis Whyte and Walter Wilcox)

Deflation is often used for QCD problems, however extremely large problems may need multigrid. Complicated approaches have been implemented [1]. Different from most multigrid, in QCD there are few grids and even the coarsest grid matrix has fairly difficult linear equations. We are testing deflating eigenvalues on this coarsest grid (and plan to try other grid levels). Because the same matrix is solved (or smoothed) many times to low accuracy, this is an ideal situation for deflation. In preliminary testing with a 2-D lattice from the Schwinger model, fine-grid-equivalent matrix-vector products are reduced from 17,027 to 2248.

2.3 Deflation for singular systems (with Jennifer Loe)

We consider singular linear equations where there are also small non-zero eigenvalues that do matter. This is different from image deblurring where small eigenvalues may need to be ignored. The null space may be roughly known in some engineering applications. We propose a method that we hope will be a break-through for the case where the null space is unknown. It computes a null vector by solving a non-singular system, then solves the linear equations with the nullity removed. Here is the two-step process that avoids singularity in both phases:

Phase 1) Apply GMRES-DR to \((A + \sigma bb^T)v = b\) to find null vector \(v\). Let \(n = v/\|v\|\).

Phase 2) Continue the GMRES-DR iteration from Phase 1 on \((I - nn^T)A(I - nn^T)x = (I - nn^T)b\). This is possible due to the special structure in Phase 1. The approximate eigenvectors developed for \(A + \sigma bb^T\) are improved as eigenvectors of \(A\) during this iteration, and they deflate eigenvalues.

Example: The matrix has diagonal elements \([0, 0, 0, 1, 2, \ldots, 9, 1, 2, 3, 4, \ldots, 1988]\) and superdiagonal elements \([0, 0, 2, 2, 2, \ldots, 2]\). The right-hand side is random. The Phase 1 iteration uses rank-one perturbation of norm 10, and it converges to residual norm of \(10^{-10}\) in 27 cycles of GMRES-DR(40,15). Then Phase 2 takes seven cycles for the deflated residual (null component taken out) to reach that level. For some comparison, RRGMRES [2] does not converge in 1000 cycles of length 40 (however, RRGMRES is not designed for such a problem with small eigenvalues that matter).

References


Verifying partial eigenvalues of Hermitian generalized eigenproblems using contour integrals

Akira Imakura, Keiichi Morikuni, and Akitoshi Takayasu

Faculty of Engineering, Information and Systems, University of Tsukuba
https://researchmap.jp/KeiichiMorikuni/

Abstract

Consider verifying all \( m \) eigenvalues \( \lambda_i \), counting multiplicity, of the Hermitian generalized eigenproblem

\[
Ax_i = \lambda_i Bx_i, \quad x_i \in \mathbb{C}^n \setminus \{0\}, \quad i = 1, 2, \ldots, m
\]

in a prescribed interval \( \Omega = [a, b] \subset \mathbb{R} \), where \( A = A^H \in \mathbb{C}^{n \times n} \), \( B = B^H \in \mathbb{C}^{n \times n} \) is positive semidefinite. We denote the eigenvalues outside \( \Omega \) by \( \lambda_i \) (\( i = m+1, m+2, \ldots, r \)), where \( r = \text{rank} B \).

To solve the eigenproblem (1), the block Sakurai–Sugiura Hankel method [1] reduces a given eigenproblem into a generalized eigenproblem of block Hankel matrices consisting of complex moments. The \( p \)th complex moment is given by

\[
M_p = \frac{1}{2\pi i} \oint_{\Gamma} (z - \gamma)^p V^H B (z B - A)^{-1} B V \, dz = \sum_{k=1}^{m} (\lambda_k - \gamma)^p \mathcal{V}_k, \quad p = 0, 1, 2, \ldots, 2M - 1
\]

deined on the closed Jordan curve \( \Gamma \) through the end points of the interval \( \Omega = [a, b] \), where \( \mathcal{V}_k = V^H B x_k x_k^H B V \in \mathbb{C}^{L \times L} \). Here, the block size \( L \in \mathbb{N}_+ \), the order of moment \( M \in \mathbb{N}_+ \), a random matrix \( V \in \mathbb{C}^{n \times L} \), and the scaling factors \( \gamma, \rho \in \mathbb{R} \) are parameters. Let \( H_M \) and \( H_M^\leq \) be block Hankel matrices whose \((i, j)\) blocks are \( M_{i+j-2} \) and \( M_{i+j-1} \), respectively. If \( \text{rank} H_M = m \), then the eigenvalues of matrix pencil \( z H_M - H_M^\leq \) and those of (1) are the same [1]. This method motivated us to develop a verification method that is efficient in parallel since the linear solves in (2) for quadrature points can be evaluated independently.

Evaluating all errors in computing the complex moments, we develop a verification method for (1). Based on the truncation errors for the trapezoidal rule of numerical quadrature for (2) [3], we derive a numerically computable enclosure of the complex moment. Then, we take numerical errors of the quadrature into account and rigorously enclose the block Hankel matrices. Taking a domain of integration \( \Gamma \) as the circle with center \( \gamma = (a + b)/2 \) and radius \( \rho = (b - a)/2 \), we approximate the complex moment (2) via the \( N \)-point trapezoidal rule with the quadrature points \( z_j = \gamma + \rho e^{i \theta_j} \), \( \theta_j = (2j - 1)\pi/N \) for \( j = 1, 2, \ldots, N \). Thus, we have the splitting \( M_p^{(N)} = M_{p,\text{in}}^{(N)} + M_{p,\text{out}}^{(N)} \), where

\[
M_{p,\text{in}}^{(N)} = \sum_{k=1}^{m} (\lambda_k - \gamma)^p \left( \frac{1}{1 - \left( \frac{\lambda_k - \gamma}{\rho} \right)^N} \right) \mathcal{V}_k, \quad M_{p,\text{out}}^{(N)} = \sum_{k=m+1}^{r} (\lambda_k - \gamma)^p \left( \frac{-\left( \frac{\rho}{\lambda_k - \gamma} \right)^N}{1 - \left( \frac{\rho}{\lambda_k - \gamma} \right)^N} \right) \mathcal{V}_k.
\]

We show that the Hankel matrix pencils \( z H_M - H_M^\leq \) and \( z H_{M,\text{in}}^{(N)} - H_{M,\text{in}}^\leq \) with \( H_{M,\text{in}}^{(N)} \) and \( H_{M,\text{in}}^\leq \) whose \((i, j)\) blocks are \( M_{i+j-1,\text{in}}^{(N)} \) and \( M_{i+j-2,\text{in}}^{(N)} \), respectively, have the same eigenvalues, if \( \text{rank}(H_M) = m \). Hence, we focus on enclosing \( M_{p,\text{in}}^{(N)} \) instead of \( M_p \). If \( \bar{M}^{(N)}_p \) is a numerical approximation of \( M^{(N)}_p \), then the computable rigorous enclosure of \( M^{(N)}_{p,\text{in}} \) is given by

\[
M_{p,\text{in}}^{(N)} \in \left( \bar{M}_p^{(N)}, \bar{M}^{(N)}_{p,\text{out}} \right) \subset \left( \bar{M}_p^{(N)}, \bar{M}^{(N)}_{p,\text{out}} \right) + \left| \bar{M}_p^{(N)} - \bar{M}^{(N)}_p \right|,
\]

281
where $(C, R)$ denotes the interval matrix with radius $R \in \mathbb{R}_{+}^{L \times L}$ and center $C \in \mathbb{C}^{L \times L}$. Thus, we obtain the entrywise bound of the complex moment (3)

$$|M_{p, \text{out}}^{(N)}| \leq (r - m)|\tilde{\lambda} - \gamma|^p \|V^H B V\|_F \left(\frac{p}{|\tilde{\lambda} - \gamma|}\right)^N \left[1 - \left(\frac{p}{|\tilde{\lambda} - \gamma|}\right)^N\right]^{-1}$$

for $p = 0, 1, \ldots, 2M - 1$ and $2M - 1 < N$, where $\tilde{\lambda}$ satisfies $|\tilde{\lambda} - \gamma| = \min_{k=m+1,m+2,\ldots,r} |\lambda_k - \gamma|$ and $\|\cdot\|_F$ denotes the Frobenius norm. This bound is interval arithmetic-friendly [4] and enables to determine the required number of quadrature points beforehand as

$$N \geq \frac{\log \frac{c + \delta}{c}}{\log \frac{p}{|\tilde{\lambda} - \gamma|}}, \quad c = (r - m)\|V^H B V\|_F \max_{p=1,2,\ldots,2M-1} |\tilde{\lambda} - \gamma|^p$$

according to the degree of precision one needs and the cost one can pay, where $\delta$ denotes the tolerance of quadrature error.

Numerical experiments on artificially generated problems show that the proposed method outperforms a standard method, and infer that the proposed method will be potentially efficient in parallel. It is also shown that the proposed method applies to practical problems.

We point out that our verification method works for multiple eigenvalues and semidefinite $B$. Each quadrature point gives rise to a linear system, and its structure enables us to develop an efficient technique to verify the approximate solution for positive definite $B$, which does not need a numerical matrix inverse. This technique breaks the problem size barrier of one million.

This work will be published in [2].

**References**


An algorithm to numerically compute the Clebsch-Gordan coefficients of unitary representations of compact groups

A. Ibort, A. López-Yela, and J. Moro

Departamento de Matemáticas, Universidad Carlos III de Madrid
https://matematicas.uc3m.es/index.php/julio-moro-carreno

Abstract

Given a compact Lie group \( G \), a finite-dimensional unitary representation of \( G \) is a pair \((\mathcal{H},U)\), where \( \mathcal{H} \) is a finite-dimensional Hilbert space and \( U : G \rightarrow U(\mathcal{H}) \) is a group homomorphism, with \( U(\mathcal{H}) \) the group of unitary operators on \( \mathcal{H} \). This representation is said to be irreducible whenever there are no proper linear invariant subspaces of \( \mathcal{H} \), i.e., when the only linear subspaces \( W \) of \( \mathcal{H} \) such that \( U(g)W \subset W \) for all \( g \in G \) are either \( W = \{0\} \) or \( W = \mathcal{H} \). In that case, \((\mathcal{H},U)\) is said to be an irreducible unitary linear representation ("irrep" in what follows).

Now, consider a closed subgroup \( H \) of \( G \): the restriction of \( U \) to \( H \) defines a unitary representation of \( H \) which is, in general, not reducible, i.e., it will possess nontrivial invariant linear subspaces. One can show that in this situation all possible irreps of \( H \) can be partitioned into equivalence classes, say \( \{\alpha : \alpha \in \hat{H}\} \), in such a way that the total Hilbert space \( \mathcal{H} \) can be decomposed as a direct sum

\[
\mathcal{H} = \bigoplus_{\alpha \in \hat{H}} \mathcal{L}^\alpha, \quad \mathcal{L}^\alpha = \bigoplus_1^{c_\alpha} \mathcal{H}^\alpha, \tag{1}
\]

i.e., each \( \mathcal{L}^\alpha \) is a proper invariant subspace associated with a different equivalence class \( \alpha \in \hat{H} \), and each \( \mathcal{L}^\alpha \) is a direct sum with itself \( c_\alpha \) times of the Hilbert space \( \mathcal{H}^\alpha \) in the irrep \((\mathcal{H}^\alpha,U^\alpha)\) of \( H \) for the class \( \alpha \). The multiplicities \( c_\alpha \) satisfy

\[
\dim \mathcal{H} = \sum_{\alpha \in \hat{H}} c_\alpha n_\alpha,
\]

where \( n_\alpha = \dim \mathcal{H}^\alpha \), and each invariant subspace \( \mathcal{L}^\alpha \) has dimension \( c_\alpha n_\alpha \). Furthermore, all unitary operators \( U(h), h \in H \) have the same block structure

\[
U(h) = \bigoplus_{\alpha} c_\alpha U^\alpha(h),
\]

where \( U^\alpha(h) \) is just the restriction to \( \mathcal{H}^\alpha \) of \( U(h) \). Thus, knowledge of the decomposition (1) above leads to a full description of the action of the operators \( U(h) \) for \( h \in H \).

The Clebsch-Gordan problem of \((\mathcal{H},U)\) with respect to the subgroup \( H \) consists in determining an orthonormal basis of \( \mathcal{H} \) adapted to the decomposition (1) above. To be more precise, given an arbitrary orthonormal basis of \( \mathcal{H} \), the entries of the matrix of change of basis from that basis to the orthonormal one adapted to (1) are called the Clebsch-Gordan coefficients of the decomposition.

Considerable effort has been put in computing the Clebsch-Gordan coefficients for various situations of physical interest. For instance, the groups SU(N) have been widely discussed (see [1] and [2] and references therein). However, all these results depend critically on the algebraic structure of the underlying group \( G \) (and the subgroup \( H \)), and no algorithm was known so far to efficiently compute the Clebsch-Gordan matrix for a general subgroup \( H \) of an arbitrary compact group \( G \).
In this talk we present an algorithm, described in detail in [3], which numerically computes the decomposition of any finite-dimensional unitary reducible representation of a compact Lie group. The algorithm, which does not rely on algebraic insight into the group structure, is inspired by quantum mechanical notions (to be more precise, by ideas in quantum tomography involving mixed quantum states, i.e., density matrices adapted to a given representation). After generating two adapted states, and after appropriate algebraic manipulations, the algorithm returns the block matrix structure of the representation in terms of its irreducible components. It also provides an adapted orthonormal basis.

The algorithm can be used to compute the Clebsch-Gordan coefficients of the tensor product of irreducible representations of a given compact Lie group. The performance of the algorithm is tested on examples like the decomposition of the regular representation of two finite groups, or the computation of Clebsch-Gordan coefficients of two examples of tensor products of representations of SU(2).

References


Task-based Algorithms and Software for Solving Dense Nonsymmetric Eigenvalue Problems

Mirko Myllykoski

Department of Computing Science and HPC2N, Umeå University
http://people.cs.umu.se/~mirkom

Abstract

This talk centers around StarNEig [1, 2, 3], a new library for solving dense nonsymmetric eigenvalue problems

\[ Ax_i = \lambda_i x_i, \]  

(standard)

\[ Ax_i = \lambda_i B x_i, \]  

(generalized)

where \( A, B \in \mathbb{R}^{n \times n} \), \( 0 \neq x_i \in \mathbb{C}^n \) and \( \lambda_i \in \mathbb{C} \). In the standard case, the route of acquiring all eigenvalues \( \lambda_i \) and the associated eigenvectors \( x_i \) of the matrix \( A \) includes the following three steps:

**Hessenberg reduction:** The matrix \( A \) is reduced to upper Hessenberg form \( H \in \mathbb{R}^{n \times n} \).

**Schur reduction:** The Hessenberg matrix \( H \) is reduced to real Schur form \( S \in \mathbb{R}^{n \times n} \).

**Eigenvectors:** The eigenvectors \( x_i \) are solved and backtransformed to the original basis.

Optionally, a fourth step can be performed to acquire a desired invariant subspace of \( A \):

**Eigenvalue reordering:** The Schur real form \( S \) is reordered, such that a selected set of eigenvalues appears in the leading diagonal block of an updated real Schur form \( \hat{S} \in \mathbb{R}^{n \times n} \).

All four steps are implemented in both LAPACK and ScaLAPACK libraries. In particular, ScaLAPACK’s carefully optimized codebase, together with its use of the two-dimensional block cyclic distribution, have been a very successful approach for solving eigenvalue problems. However, ScaLAPACK does not perform particularly well on modern heterogeneous hardware, that consists of both high core count CPUs and GPUs, due to its static and synchronous nature.

StarNEig relies on a more modern approach of describing the algorithms as acyclic directed graphs known as task graphs, where the vertices represent the various computational operations (a.k.a tasks) and the edges describe the data dependencies between the tasks. A runtime system (StarPU [4]) is responsible for scheduling the tasks to the various computational resources, such as CPU cores and GPUs, in a sequentially consistent order as dictated by the data dependencies. This leads to more flexible algorithms that are able to adapt to different input matrices and modern computing hardware. Central to this improvement is the fact that carefully selected and constructed task graphs enclose many opportunities for increased concurrency which can be automatically detected and exploited by the runtime system. A task-based algorithm can thus reach a more efficient resource utilization than the corresponding ScaLAPACK algorithm.

StarNEig aims to provide a complete task-based software stack for solving dense nonsymmetric eigenvalue problems. Currently, StarNEig implements the whole software stack for standard eigenvalue problems in shared memory. Support for distributed memory is currently a work in progress but both the Schur reduction and eigenvalue reordering steps are fully operational in both shared
and distributed memory. The situation is similar with generalized eigenvalue problems with the exception that the Hessenberg-triangular step is currently under development. The missing software components are implemented as LAPACK and ScaLAPACK wrapper functions. Future work with StarNEig includes the implementation and integration of the missing software components. Support for complex valued matrices is also planned.

StarNEig is significantly faster than ScaLAPACK when it comes to the Schur reduction and eigenvalue reordering steps. In particular, StarNEig is almost three times faster when computing a real Schur form and up to five times faster when reordering a real Schur form, as demonstrated in [3]. The main reason for this much improved performance is the fact that the task-based approach has eliminated various synchronization points, and thus, different computational steps are allowed to overlap and merge together. This has significantly improved the resource utilization since low priority tasks can be delayed until the computational resources start becoming idle. In addition, StarNEig has been demonstrated to scale reasonably well when computing a real Schur form and almost linearly when reordering a real Schur form. Preliminary experiments have also demonstrated that StarNEig can indeed take advantage of the available GPU resources. However, the GPU support, and the multi-GPU support in particular, are still under active development.

In addition to providing higher performance, StarNEig contains functionality that is currently missing from ScaLAPACK. In particular, the components of StarNEig that compute the standard and generalized eigenvectors are robust. That is, the computed eigenvectors are always in the representable range of double precision floating-point numbers. Although same functionality exists in LAPACK, the implementation in StarNEig is both robust and tiled, the latter of which lead to a parallel implementation with level 3 BLAS performance. The work on this part of StarNEig has been done by C.C. Kjelgaard Mikkelsen, A. Schwarz and L. Karlsson (see [5, 6]).

StarNEig has been developed by several people at Umeå University: Mirko Myllykoski, Carl Christian Kjelgaard Mikkelsen, Angelika Schwarz, Lars Karlsson, and Bo Kågström. The work has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 671633 (NLAFET). Support has also been received from eSSENCE, a collaborative e-Science programme funded by the Swedish Government via the Swedish Research Council (VR).

References


Deflation and Multigrid: Similarities and Differences

Reinhard Nabben

Institute for Mathematics, TU Berlin
www.math.tu-berlin.de/~nabben/

Abstract

Among the best preconditioners for solving linear systems of equations are multilevel and multigrid methods. Alternatively, deflation or augmentation techniques can be used to speed up convergence of the Krylov subspace method. Originally, multigrid and deflation arise from different areas of numerical analysis.

Nevertheless, deflation methods are closely related to multigrid or multilevel and domain decomposition methods. We will show that all these methods can be seen as subspace correction methods. Moreover, we will introduce a unified framework that can be used to analyze these methods simultaneously.

Another common ingredient are projections. Both type of methods make use of projections. In deflation methods the original system is multiplied by a projection which leads to a singular linear system whose spectrum might be more favorable for the Krylov subspace method. On the other hand the coarse grid correction in multigrid is a projection. But projections are also used to analyze both type of methods.

Here, we prove new theorems for projections. We describe the spectrum of the product of an arbitrary matrix with a projection in comparison to the spectrum of the product added by the complementary projection. This surprising coupling between the spectra allows not only elegant proofs for the relation between deflation and multigrid methods but also highlight the role of projections in this area. With this new theorem we generalize several known results. We are able to give short proofs for the characterization of the eigenvalues of the multigrid preconditioner given in the literature.

Moreover, we present a new characterization not only of the eigenvalues of the (preconditioned) deflated system but also of the multigrid preconditioned system. This result holds for non-symmetric matrices. But it is also new for symmetric matrices.

Applied to symmetric positive definite systems our new characterization of the eigenvalues of the multigrid preconditioner leads to a simple approach to the many results for algebraic multigrid methods. Short and elegant proofs can be given. For example we can determine the optimal interpolation operator in the multigrid method that gives the smallest spectral radius of the iteration matrix. Moreover, we can show that the found optimal interpolation operators are also optimal with respect to the condition number of the multigrid preconditioned system.

Several other results, e.g. bounds for the eigenvalues of the multigrid preconditioner, can be obtained easily using our new approach.

The talk is partially based on [1] and [2]

References

Partial Nonnegative Matrix Factorization in Fluorescence Microscopy

James Nagy and Blair Rossetti

Mathematics Department, Emory University
www.mathcs.emory.edu/~nagy

Abstract

The standard nonnegative matrix factorization (NMF) problem is simple to state:

Given a nonnegative matrix $Y$, find nonnegative matrices $S$ and $W$ such that $Y \approx SW$.

This problem arises in many applications, and is generally used to select features that are hidden in the data matrix $Y$. Although the NMF problem is simple to state, it poses many computational challenges (such as non-uniqueness) that are usually addressed by including additional, application dependent, constraints and/or penalty terms on $S$ and $W$.

In this work we consider the partial nonnegative matrix factorization problem with an $\ell_1$ regularization term:

$$\min_{S_A, W, b} \| Y - [S_0 | S_A] W + b1^T \|_F + \gamma \|W\|_1$$

subject to $S_A, W, b \succeq 0$ (1)

where $Y$ and $S_0$ (i.e., some columns of $S$) are given, $1$ is a vector of ones, and the symbol $\succeq$ is used to mean the elements of the matrices $S_A$ and $W$, and the elements in the vector $b$ are nonnegative.

The aim of this poster is to describe an application in fluorescence microscopy where this problem arises, and to describe some numerical approaches to compute approximations.

References

Symmetric eigenvalue perturbation in the tangent space

Yuji Nakatsukasa
Mathematical Institute, Oxford University
https://people.maths.ox.ac.uk/nakatsukasa

Abstract
Among the most classical and important results in eigenvalue perturbation theory is Weyl’s theorem: for \( n \times n \) symmetric (or Hermitian) matrices \( A \) and \( E \),
\[
|\lambda_i(A + \epsilon E) - \lambda_i(A)| \leq \epsilon \|E\|_2. \tag{1}
\]
This shows in particular that eigenvalues of symmetric matrices are always well conditioned.

Now when a (not necessarily symmetric) matrix \( A \) is perturbed by its commutator \( A + \epsilon(AB - BA) \), the perturbation of semisimple eigenvalues is in the order of \( O(\epsilon^2) \) rather than \( O(\epsilon) \). To see this, note that
\[
A + \epsilon(AB - BA) = (I - \epsilon B)A(I + \epsilon B) + \epsilon^2 B^2
\]
and since
\[
(I - \epsilon B)(I + \epsilon B) = I - \epsilon B^2,
\]

it follows that \( (I - \epsilon B)A(I + \epsilon B) \) is a similarity transformation of \( A \) up to an \( O(\epsilon^2) \) term. Put another way, the set of commutators \( \{AX - XA : X \in \mathbb{R}^{n \times n}\} \) lies in the tangent space to the manifold of matrices similar to \( A \). This is the principle underlying Lax pairs [?], which plays an important role e.g. in mechanics.

For symmetric matrices, this observation indicates that Weyl’s bound in (1) can be improved to \( O(\|E\|_2^2) \) if \( E \) can be written as a commutator of \( A \). In this work we show that for symmetric \( A \) and skew-symmetric \( B \) (so that \( AB - BA \) is symmetric)
\[
|\lambda_i(A) - \lambda_i(A + \epsilon(AB - BA))| \leq \epsilon^2 \|B\|_2^2 \|AB - BA\|_2. \tag{2}
\]

Important contributions in this direction include [?] and [?], which deal with perturbing block diagonal matrices by off-diagonal perturbations. Mathias [?] shows that
\[
\left| \lambda_i \left( \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \right) - \lambda_i \left( \begin{bmatrix} A_1 & \epsilon E^T \\ \epsilon E & A_2 \end{bmatrix} \right) \right| \leq \frac{\epsilon^2 \|E\|^2}{\text{gap}}, \tag{3}
\]

where gap is the spectral gap between the eigenvalues of \( A_1 \) and \( A_2 \). In a beautiful paper, Li and Li [?] obtain the slightly tighter bound
\[
\left| \lambda_i \left( \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \right) - \lambda_i \left( \begin{bmatrix} A_1 & \epsilon E^T \\ \epsilon E & A_2 \end{bmatrix} \right) \right| \leq \frac{2\epsilon^2 \|E\|^2}{\text{gap} + \sqrt{\text{gap}^2 + 4\epsilon^2 \|E\|^2}}. \tag{4}
\]

Note that when gap \( > 0 \) in the above settings (3), (4), the off-diagonal perturbation lies in the tangent space to the manifold of matrices similar to \( A \). Indeed, our bound (2) can be used to establish Mathias’s bound (3) as a special case.

More generally, one can split an arbitrary perturbation \( E \) into a sum of a commutator and another matrix as \( E = (AB - BA) + W \). We can then bound the eigenvalue perturbation as
\[
|\lambda_i(A) - \lambda_i(A + \epsilon E)| \leq \epsilon^2 \|B\|_2 \|AB - BA\|_2 + \epsilon \|W\|_2.
\]

For example when \( A = U \Lambda U^T \) has distinct eigenvalues, \( AB - BA \) is such that the diagonal elements of \( U^T(AB - BA)U \) are all zero, and \( W \) can be of the form \( UDU^T \) where \( D \) is diagonal.
Towards a robust eigensolver for large sparse nonlinear eigenproblems

Stefan Güttel, Gian Maria Negri Porzio, and Françoise Tisseur

Department of Mathematics, The University of Manchester
https://gmnp.bitbucket.io/

Abstract

Given a nonempty, open, bounded set $\Omega \subset \mathbb{C}$ and a matrix-valued function $F: \Omega \rightarrow \mathbb{C}^{n \times n}$, the nonlinear eigenvalue problem (NEP) consists of finding all the eigenpairs $(\lambda, v) \in \Omega \times \mathbb{C}^n \setminus \{0\}$ such that

$$F(\lambda)v = 0.$$  \hspace{1cm} (1)

Nonlinear eigenvalue problems appear in many areas of computational science and engineering, such as control theory, fluid mechanics and structural engineering [5].

Even though there is yet no clear best algorithm to solve NEPs, every NEP solver broadly falls into three categories: solvers based on Newton’s method, solvers based on contour integration, and solvers based on linearization [5]. Our interest is on the latter class of eigensolvers. These eigensolvers start by constructing a polynomial or rational approximation $R(\lambda)$ of the matrix-valued function $F(\lambda)$ on $\Omega$ so as to replace the NEP (1) with the approximate eigenproblem,

$$R(\lambda)v = 0,$$  \hspace{1cm} (2)

which is still nonlinear in $\lambda$ but simpler to handle as long as $R(\lambda)$ is expressed in appropriate bases. Then (2) is rewritten as a linear eigenproblem $L(\lambda)x = 0$ of larger dimension—the process of linearization. Finally, the linear eigenproblem is solved and approximate eigenpairs $(\lambda, v)$ of $F(\lambda)$ are recovered from approximate eigenpairs of $L(\lambda)$. The quality and robustness of such eigensolvers depend on a good understanding and careful implementation of these steps. Our main focus in this talk is on the rational approximation step.

There is already a large body of work on the approximation or interpolation of matrix-valued functions. Taylor approximation has been applied successfully, in particular to NEPs arising in delay differential equations [8]. Effenberger and Kressner focused their attention on the Chebyshev interpolants, which is most appropriate if the wanted eigenvalues lie on a smooth curve in the complex plane [3]. Van Beeumen, Michiels and Meerbergen suggested a linearization for polynomial eigenvalue problems based on Hermite and Lagrange bases [11]. Güttel et al. proposed to use a rational Leja-Bagby sampling approach to construct $R(\lambda)$ in rational Newton basis [6].

Recently, Nakatsukasa, Sète and Trefethen proposed the Adaptive Antoulas–Anderson (AAA) algorithm to approximate scalar functions $f: \Omega \rightarrow \mathbb{C}$ by rational functions in barycentric form [10]. The AAA algorithm is robust and flexible: it usually considers fine mesh grids $Z$ of $\Omega$, therefore it works well with disconnected regions or regions with irregular shapes. Furthermore, its implementation and usage are quite simple: once the user gives the set $Z$ of the initial points and a threshold $\varepsilon$, the AAA algorithm returns a rational approximant $r(z)$ such that $|f(z) - r(z)| < \varepsilon$ for every $z \in Z$.

The flexibility and the easiness of implementation led many authors to focus on generalizing the AAA to matrix-valued functions. Güttel and Elsworth proposed to approximate a probing function $r(z) \approx f(z) := v_1^TF(z)v_2$, where the $v_i$ are random vectors, and then use the sample points of $r(z)$ to build the matrix approximant $R(z)$ [4]. Lietaert et al. proposed the set-valued AAA, where they
Moreover, we propose a hybrid approach when the explicit form of an analytic function $F(z)$ is given and they approximate each $f_k(z)$ at the same time [9]. In our work we mainly propose two variations. First, we add a normalization to the set-valued AAA that also considers the norms of the coefficients $\|A_k\|$. In their original work, the goal is approximating the scalar functions such that $\max_{z \in Z} |f_k(z) - r_k(z)| < \varepsilon$, where $Z \subset \Omega$ is a finite set, and $f_k(z)$ are normalized such that $\max_{z \in Z} |f_k(z)| = 1$. We suggest that the algorithm should consider

$$ \max_{z \in Z} |f_k(z) \cdot \|A_k\| - r_k(z)| < \varepsilon \cdot \|F\|_Z $$

to build $R(z) = \sum_{k=1}^s r_k(z)A_k$, with $\|F\|_Z := \sup_{z \in Z} \|F(z)\|_2 = \max_{z \in Z} \|F(z)\|_2$. This choice gives better theoretical conclusions and it usually produces better approximations, while giving the same numerical results in the worst case scenario. In fact, the final error estimate becomes $\|F - R\|_Z \leq s\varepsilon \|F\|_Z$. In addition, if we consider the backward error for an eigenpair $(\lambda, v)$ of $R(z)$

$$ \eta(\lambda, v) := \min\{\tilde{\varepsilon} : (F(\lambda) + \Delta F(\lambda))v = 0, \|\Delta F\|_\Omega \leq \tilde{\varepsilon}\|F\|_\Omega\} = \frac{\|F(\lambda)v\|_2}{\|F\|_\Omega \|v\|_2} \leq \frac{\|F(\lambda)v\|_2}{\|F\|_Z \|v\|_2}, $$

then we can bound it by

$$ \frac{\|F(\lambda)v\|_2}{\|F\|_Z \|v\|_2} = \frac{\|(F(\lambda) - R(\lambda))v\|_2}{\|F\|_Z \|v\|_2} \leq \frac{\|F - R\|_Z}{\|F\|_Z} \leq s\varepsilon. $$

Moreover, we propose a hybrid approach when the explicit form of an analytic function $F(z)$ is not available. In this case, instead of only relying on the approximation $r(z)$ of the probing function $f(z)$, we extract the zeros and the poles of $r(z)$ and then we use this information to build a condenser for a Leja-Bagby approximation [12], [1], [5]. This approach reaches better results than the ones showed in [4] for functions that are analytic inside $\Omega$. In addition, it allows us to give a theoretical justification of the exponential decay of the approximation error of $R(z)$.

Finally, we present many numerical experiments where we compare all these algorithms on different nonlinear eigenvalue problems [2], [7]. These experiments will be publicly available and will allow other researchers to explore new ideas on rational approximants.

References


Efficient Training of Neural Networks via Variable Projection

Elizabeth Newman, Joseph Hart, Bart van Bloemen Waanders, and Lars Ruthotto

Department of Mathematics, Emory University
http://math.emory.edu/~enewma5/

Abstract

To take advantage of the surplus of available data and the substantial computational power, it is essential to develop efficient, data-driven algorithms. Neural networks are perhaps the most ubiquitous of data-driven tools, highly successful in, e.g., image classification [6], dimensionality reduction [3], and reinforcement learning [4]. These flexible models are often trained by optimizing

\[
\min_{\theta, W} \Phi(\theta, W) \equiv \frac{1}{|D|} \sum_{(y, c) \in D} L(Wf(y, \theta), c) + R(\theta) + S(W),
\]

where \( L \) is a convex loss function (in the first argument) and \( R \) and \( S \) are convex regularizers. The training data \((y, c) \in D \subset \mathbb{R}^n \times \mathbb{R}^m\) consists of known pairs of input-output features. The neural network \( f \) is parameterized model defined as a sequence of simple, nonlinear layers. The trainable parameters in (1) are the network weights \( \theta \) and a dense matrix \( W \).

Training a neural network (i.e., updating the parameters) is challenging due to, e.g., the presence of local minima and ill-conditioning of the objective function [3, 1]. In this talk, we will show this ill-conditioning by analyzing the Hessian matrix of \( \Phi \) with respect to the learnable parameters. To address these challenges, neural networks are typically trained using stochastic gradient descent (SGD), which has its own drawbacks such as slow convergence, difficult parallelization, and significant hyperparameter tuning [1]. Second-order methods can overcome these limitations, but are difficult to implement for stochastic approximations and often stagnate at suboptimal local minima for stochastic average approximations [3].

In this talk, we consider non-stochastic, second-order methods to train neural networks efficiently and to a high-level of accuracy. For many applications, such as high-dimensional surrogate modeling [9], the quality of the neural network approximation is of primary importance. To train highly-accurate neural network models, we reformulate (1) using variable projection (VarPro)

\[
\min_{\theta} \Phi(\theta, W(\theta)) \quad \text{s.t.} \quad W(\theta) = \arg\min_W \Phi(\theta, W).
\]

The VarPro model has been well-studied for separable, nonlinear least squares problems [2] (i.e., a least squares loss function in (1)). Finding the optimal \( W(\theta) \) in (2) involves solving a least squares problem at every training iteration which can be done efficiently via SVD or QR factorizations [7, 5]. In the context of neural networks, training with VarPro can be more efficient due to the reduced parameter space [7] and can be shown to improve the conditioning [8]. In addition, the implementation of VarPro depends only on the number of output features, which is typically small and, more importantly, independent of number of parameters \( \theta \) and the network architecture \( f \). Thus, the overhead for training with VarPro is minimal.

In this talk, we will present VarPro for general convex objective functions \( \Phi \), extending the algorithm from separable, nonlinear least squares to, e.g., multimodal logistic regression which is popular for classification tasks. To generalize VarPro, we will derive the Jacobian of the weights \( W \) with respect to \( \theta \) using implicit differentiation. We require this Jacobian to perform backward propagation, a
gradient-based algorithm through which we update the parameters $\theta$. Once we have this Jacobian, we will compare the performance of various optimization methods, such as SGD, Gauss-Newton, and BFGS, with and without VarPro. Through these numerical experiments, we will demonstrate the benefits of VarPro on the convergence rate and conditioning. Time permitting, we will extend this VarPro algorithm further to compute the full Hessian through which we can implement and analyze VarPro for the second-order Newton’s method.

References


Fast implementation of the traveling-salesman-problem method for reordering columns within supernodes in sparse Cholesky factorization

Mathias Jacquelin, Esmond G. Ng, and Barry W. Peyton

Computational Research Division, Lawrence Berkeley National Laboratory
https://crd.lbl.gov/about/staff/director/esmond-ng/

Abstract

Consider the Cholesky factorization of an $n$ by $n$ sparse, symmetric, positive definite matrix $A$. We assume that $A$ has been symmetrically permuted to reduce fill by some ordering heuristic, such as minimum degree or nested dissection. It is well known that in most practical problem settings, the columns of the Cholesky factor $L$ are partitioned into sets of columns sharing the same sparsity structure. Each such set of columns corresponds to a so-called supernode. The sparsity pattern of a 9 by 9 Cholesky factor $L$ with three supernodes is shown on the left in Fig. 1. The first supernode $J_1 = \{1, 2\}$ comprises columns 1 and 2, the second supernode $J_2 = \{3, 4\}$ comprises columns 3 and 4, and the third supernode $I = \{5, 6, 7, 8, 9\}$ comprises columns 5 through 9.

We see that the columns within the same supernode are numbered consecutively. Furthermore, columns within the same supernode can be numbered in any order without changing the number of nonzeros in the factor matrix (with the corresponding rows being reordered symmetrically). On certain machines—particularly those using GPUs—it is desirable to exploit this freedom to reorder columns within supernodes in order to improve factorization efficiency. Loosely speaking, what is desired is to reorder the columns within supernodes so that the nonzeros of the factor matrix are organized within fewer and larger dense blocks. The number of nonzeros in the factor matrix remains the same. For example, when supernode $I = \{5, 6, 7, 8, 9\}$ in $L$ in Fig. 1 is reordered so that $I = \{6, 8, 5, 7, 9\}$, we obtain the Cholesky factor $\hat{L}$, which is shown on the right in Fig. 1. Under the new ordering of $I$, there is only one dense block joining $J_1$ to $I$ and only one dense block joining $J_2$ to $I$. This is the minimum number of such blocks over all possible reorderings of supernode $I$.

Pichon, Faverge, Ramet, and Roman [3] worked directly on the optimization problem suggested by our simple example: namely, the problem of reordering the columns within all of the supernodes to minimize the number of dense blocks linking one supernode to another. They were able to reformulate the problem as a set of traveling salesman problems (TSP), one for each supernode. Now, it is well know that TSP is an NP-hard problem. Because there are heuristics to solve their...
instance of TSP to within a factor of two, Pichon et al. were able to reduce the total number of blocks to within a factor of two of optimal. The problem with their approach is its cost in time, and the primary bottleneck is the time required to compute the “distances” for the TSPs. The strength of their method, however, is that it is very effective at reducing the total number of blocks. In our work here, we introduce three new techniques to use in computing the TSP “distances” that dramatically reduce the cost in time for their method. We discuss each of these briefly below.

First, for the TSP-based reordering method the “distance” separating two columns $i$ and $j$ taken from the same supernode $I$ is of the following form:

$$|X[i] \setminus Y[j]| + |Y[j] \setminus X[i]|,$$

(1)

where $X[i]$ and $Y[j]$ are sets of supernodes associated with $i$ and $j$, respectively. The improvement is a simple way to reduce the size of the TSP. The “distance” separating columns $i$ and $j$ is zero if and only if $X[i] = Y[j]$. Furthermore, if the distance separating $i$ and $j$ is zero, then they represent two visits to the same “city” in the TSP. Only one such “city” must be included in the TSP, thereby reducing the size of the TSP. Including a single representative column is done systematically with all of the columns of the supernode. After the reduced TSP is solved (approximately), the columns excluded from the TSP are added to the tour beside their representative column, i.e., beside the single column that is “distance” zero away and was included in the TSP.

Second, let $|I|$ be the number of columns in supernode $I$. The code supplied to us by Pichon et al. computes the “distance” in (1) in a straightforward manner; it uses $|I|^2$ passes through sets, member by member, in order to compute the required distances for the supernode. We propose to replace the expression in (1) with the following equivalent expression

$$|X[i]| + |Y[j]| - 2|X[i] \cap Y[j]|,$$

(2)

and then it is straightforward to reduce the number of passes through sets to $|I|(|I| + 1)/2$.

The third improvement grows out of work presented in Gilbert, Ng, and Peyton [1] and takes further advantage of the expression in (2) for the “distances”. The work in [3] did not exploit any structure in the sets $X[i]$ and $Y[j]$. They are treated as arbitrary and unrelated sets (of supernodes), thus requiring the processing one-by-one of the members of the sets. These sets, however, do have structure that we can exploit; the sets $X[i], Y[j]$, and also the sets $X[i] \cap Y[j]$ induce subtrees in the supernodal elimination tree of the sparse matrix $A$. We use the techniques introduced in [1] to exploit this structure; more specifically, we use those techniques to partition the relevant subtrees into paths, each of whose length can be computed in constant time once it is detected. This eliminates the need to count members of sets one-by-one.

Combining these improvements, we have observed speed-ups ranging from approximately three to six over the code we received from Pichon et al. We will compare its performance with our competitive method for this problem based on partition refinement [2] introduced in 2018. With the new improvements, the TSP-based approach to this problem becomes a much more practical alternative than it was previously.

References


Tensor Completion and its Applications

Michael K. Ng

Department of Mathematics, The University of Hong Kong
https://hkumath.hku.hk/~mng/

Abstract

The problem of recovering a low-rank tensor from an incomplete or missing sampling of its entries, also known as low-rank tensor completion, arises in a wide range of application areas, e.g., hyperspectral image processing [7, 16, 23], computer vision [13, 26], and machine learning [18, 19]. The goal of low-rank tensor completion is to recover a tensor with the lowest rank based on observable entries of a given tensor \( \mathcal{M} \) indexed by \( \Omega \), which can be expressed as follows:

\[
\min_{\mathcal{Z}} \text{rank}(\mathcal{Z}) \quad \text{s.t.} \quad \mathcal{P}_\Omega(\mathcal{Z}) = \mathcal{P}_\Omega(\mathcal{M}),
\]

where \( \Omega \) is a subset of \( \{1, \ldots, n_1\} \times \{1, \ldots, n_2\} \times \{1, \ldots, n_3\} \) and \( \mathcal{P}_\Omega \) is the projection operator such that the entries in \( \Omega \) are given while the remaining entries are missing, i.e.,

\[
(\mathcal{P}_\Omega(\mathcal{Z}))_{ijk} = \begin{cases} 
\mathcal{Z}_{ijk}, & \text{if } (i, j, k) \in \Omega, \\
0, & \text{otherwise}.
\end{cases}
\]

In particular, if \( n_3 = 0 \), the tensor completion problem (1) reduces to the well-known matrix completion problem, which has received a considerable amount of attention in the past decades, see, e.g., [2, 3, 4, 5, 17] and references therein. In the matrix completion case, a given incoherent \( n \times n \) matrix could be recovered with high probability if the uniformly random sample size is of order \( O(rn \log n) \), where \( r \) is the rank of the given matrix. This bound has been shown to be optimal, see [4] for detailed discussions.

The main aim of this talk is to study incoherence conditions of low-rank tensors under transformed tensor singular value decomposition and provide a lower bound on the number of random sample entries required for exact tensor recovery. Different kinds of tensor rank lead to different convex relaxation models and different sample sizes required for exact recovery. For example, when the rank in the model (1) is chosen as the Tucker rank [21], Liu et al. [13] proposed to use the sum of the nuclear norms (SNN) of unfolding matrices of a tensor to recover a low Tucker rank tensor. Under the SNN framework, Tomioka et al. [20] proved that a given \( m \)-th order tensor \( \mathcal{X} \in \mathbb{R}^{n \times n \times \cdots \times n} \) with Tucker rank \((r, r, \ldots, r)\) can be exactly recovered with high probability if the Gaussian measurements size is of order \( O(rn^{m-1}) \). Later, Mu et al. [15] proved that \( O(rn^{m-1}) \) Gaussian measurements are necessary for a reliable recovery under the SNN framework. In fact, the number of degrees of freedom of a tensor \( \mathcal{X} \in \mathbb{R}^{n \times n \times \cdots \times n} \) with Tucker rank \((r_1, r_2, \ldots, r_m)\) is \( \prod_{i=1}^{m} r_i + \sum_{i=1}^{m} (r_in - r_i^2) \), which is much smaller than \( O(rn^{m-1}) \). This implies that there exists room for improvement for tensor completion. Recently, Yuan et al. [24, 25] showed that an \( n \times n \times n \) tensor with Tucker rank \((r, r, r)\) can be recovered exactly with high probability with as few as \( O((r^2 n^2 + r^2 n) \log^2 (n)) \) entries observed. These sample size requirement drastically improves those based on unfolding methods which typically require a sample size of the order given in [15]. Later, Xia et al. [22] showed that an \( n \times n \times n \) tensor with multilinear rank \((r, r, r)\) can be reconstructed with high probability from as few as \( O(n^2 \log^2 n + r^2 n \log^6 n) \) entries by a gradient descent algorithm. When the rank in the model (1) is chosen as the CANDECOMP/PARAFAC (CP) rank [11], Mu et al. [15] introduced...
a square deal method which only uses an individual nuclear norm of a balanced matrix instead of using a combination of all $m$ nuclear norms of unfolding matrices of the tensor. Under this setting, they also proved that $O(r^{\lceil \frac{m}{2} \rceil} n^{\lceil \frac{m}{2} \rceil})$ samples are sufficient to recover a CP rank $r$ tensor with high probability.

The tubal rank of a third-order tensor is proposed by Kilmer et al. [9, 10]. The tubal rank is based on tensor-tensor product (t-product) and the associated algebraic framework allows tensor factorizations like matrix factorizations. This new perspective has endowed multidimensional data arrays with an advantageous representation in real-world applications. Zhang et al. [27] used the tensor tubal nuclear norm (TNN) as a convex relaxation of the tensor tubal rank, and then proved that a low tubal rank tensor can be exactly recovered by $O(r n^2 \log(n^2))$ uniformly sampled entries. However, the TNN is not the convex envelope of the tubal rank which may lead to more sample entries needed to exactly recover the original tensor. In Table 1, we summarize existing tensor completion results for $n \times n \times n$ third-order tensors. It is interesting to note that there are other factors that affect sample sizes required, e.g., sampling methods and incoherence conditions. For detailed discussions, we refer to the references [1, 8, 12, 14].

Table 1: A summary of sample sizes and sampling methods for tensor completion.

<table>
<thead>
<tr>
<th>Rank Assumption</th>
<th>Sampling Method</th>
<th>Incoherent and Other Conditions</th>
<th>Requirement Sampling Sizes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP-rank $r$ [15]</td>
<td>Gaussian</td>
<td>N/A</td>
<td>$O(r n^2)$</td>
</tr>
<tr>
<td>CP-rank $r$ [8]</td>
<td>Uniformly Random</td>
<td>Incoherent condition of symmetric tensor</td>
<td>$O(n^2 r^3 \log(n))$</td>
</tr>
<tr>
<td>Tucker rank [25]</td>
<td>Uniformly Random</td>
<td>Matrix Incoherent condition on model-n unfolding</td>
<td>$O(r n^2 + r^3 n) \log^2(n)$</td>
</tr>
<tr>
<td>Tucker rank [6]</td>
<td>Random</td>
<td>Matrix Incoherent condition on model-n unfolding</td>
<td>$O(r n^2 \log^2(n))$</td>
</tr>
<tr>
<td>Tucker rank [15]</td>
<td>Gaussian</td>
<td>N/A</td>
<td>$O(r n^2)$</td>
</tr>
<tr>
<td>Tucker rank [22]</td>
<td>Uniformly Random</td>
<td>Matrix Incoherent condition on model-n unfolding</td>
<td>$O(r^2 n^2 \log^2 n + r^7 n \log^6 n)$</td>
</tr>
<tr>
<td>Tubal rank [27]</td>
<td>Uniformly Random</td>
<td>Tensor Incoherent condition</td>
<td>$O(r n^2 \log(n^2))$</td>
</tr>
<tr>
<td>Multi-rank $(r_1, r_2, r_3)$ (this talk)</td>
<td>Uniformly Random</td>
<td>Tensor Incoherent condition</td>
<td>$O(\sum r_i n \log(n^2))$</td>
</tr>
</tbody>
</table>

In this talk, we study tensor completion for third-order tensors of size $n_1 \times n_2 \times n_3$ and investigate into incoherence conditions of $n_3$ low-rank matrix slices of sizes $n_1 \times n_2$ under the transformed tensor singular value decomposition. We show that such low-rank tensors can be recovered exactly with high probability when the number of randomly observed entries is of order $O(r \max\{n_1, n_2\} \log(\max\{n_1, n_2, n_3\}))$, where $r$ is the sum of the ranks of these $n_3$ matrix slices in the transformed tensor. We also present several synthetic and imaging data sets by using different transformations are performed to demonstrate that our theoretical result is valid and the performance of the proposed method is better than that of existing methods in terms of sample sizes requirement. Several related imaging applications are also presented for illustration.

References
References


Treating Ensemble Sample Error Covariances in Data Assimilation

Nancy K Nichols, Polly J Smith and Amos S. Lawless

Department of Mathematics and Statistics, University of Reading
www.reading.ac.uk/maths-and-stats/about/team/nancy-nichols.aspx

Abstract

High dimensional covariance matrices arise in applications from numerical weather prediction to finance. In data assimilation observations and numerical model predictions of a dynamical system are combined, weighted by their respective error covariances, to obtain improved estimates of the current (and future) states of the system. The maximum a posteriori Bayesian estimate of the states, given a prior prediction and observations over time, is obtained as the solution to a very large nonlinear least-squares variational problem. Information derived from an ensemble of model simulations is used to determine the a priori error covariances required in the assimilation. Due to restrictions on sample size, ensemble covariances are routinely rank deficient and/or ill-conditioned and marred by sampling noise; thus they require some level of modification before they can be used in a standard assimilation framework. Here, we compare methods for improving the rank and conditioning of multivariate sample error covariance matrices.

The first method, model state-space localisation via the Schur product, effectively removes sample noise, but can dampen small cross-correlation signals. The second method, reconditioning, alters the matrix eigenvalues directly; this preserves the correlation structures but does not remove sampling noise. We show it is better to recondition the correlation matrix rather than the covariance matrix, as this prevents small but dynamically important modes from being lost. A combination that exploits the merits of each method is found to offer an effective alternative. Results are presented in the context of coupled atmosphere-ocean data assimilation, where the particular challenge is to identify the cross covariances linking the errors in the atmosphere and ocean variables.
Inertia laws and localization of real eigenvalues for generalized indefinite eigenvalue problems

Yuji Nakatsukasa, and Vanni Noferini

Department of Mathematics and Systems Analysis, Aalto University, Finland
http://math.aalto.fi/~noferiv1/

Abstract

The inertia of a Hermitian matrix $A = A^* \in \mathbb{C}^{n \times n}$ is the triple

$$(n_+(A), n_0(A), n_-(A)) \in \mathbb{N}^3$$

where $n_+(A), n_0(A)$ and $n_-(A)$ are, respectively, the number of positive, zero, and negative eigenvalues of $A$. In particular, the inertia must satisfy the constraints $n_+(A) + n_-(A) + n_0(A) = n$ and $n_0(A) = n - \text{rank}(A)$.

Sylvester’s law of inertia [8] is a very classical result that states that the equivalence classes determined by a given inertia coincide with the orbits by congruence:

**Theorem 1 (J. J. Sylvester, 1852)** Let $A = A^*, B = B^* \in \mathbb{C}^{n \times n}$. Then $A$ and $B$ have the same inertia if and only if there exists an invertible $X \in \text{GL}(n, \mathbb{C})$ such that $A = X^*BX$.

The law of inertia is still a useful tool in many applications, such as counting the number of eigenvalues of a Hermitian matrix in a given interval, which is helpful in the design of algorithms that compute eigenvalues or singular values with high relative accuracy [2, Chapter 5].

Some generalization of Sylvester’s original theorem are known: Ikramov [4] proved that two normal matrices are congruent if and only if they have the same number of eigenvalues on any semiline starting from 0 in the complex plane. Kostić and Voss studied Sylvester-like laws of inertia for nonlinear eigenvalue problems [6], but limited to the case where minmax characterization of the eigenvalues exist: this excludes several linear and nonlinear problems of practical interest; in particular, this excludes indefinite generalized eigenvalue problems. Bilir and Chicone [1] investigated the localization of the real parts of the eigenvalues of quadratic matrix polynomials with certain assumptions on the coefficients.

In this talk, I plan to discuss an extension of Sylvester’s classical theorem in a different direction than what was done in the above mentioned literature. I will analyze Hermitian indefinite generalized, polynomial, and generally nonlinear eigenvalue problems. I will argue that, just from the inertiae of the coefficients, generally it is not possible to determine exactly the number of positive, zero, negative, and overall real eigenvalues of the eigenvalue problem. However, what is possible is to obtain some nontrivial bounds on these quantities that only depend on the inertiae of some appropriate coefficients, and as such are invariant by congruence transformations of them.

One particularly interesting case is that of linear generalized eigenvalue problems, associated with a pencil $A - zB$ where $A = A^*, B = B^* \in \mathbb{C}^{n \times n}$. As I will discuss, the closer (in terms of fraction of eigenvalues having the same sign) one of the coefficient matrices is to definiteness, the sharper the bounds become.

As an example of the results that can be obtained and that I will present, below is a bound on the number of positive eigenvalues of the pencil.
\textbf{Theorem 2} Let $A, B \in \mathbb{C}^{n \times n}$ be Hermitian and have inertia, respectively $(n_+(A), n_0(A), n_-(A))$ and $(n_+(B), n_0(B), n_-(B))$. Then the number $n_+(A, B)$ of positive eigenvalues of the pencil $A - zB$ satisfies
\[ N_{++} - n \leq n_+(A, B) \leq 2n - |n_0(A) - n_0(B)| - N_{+-} \]
where $N_{++} = \max\{n_+(A) + n_+(B), n_-(A) + n_-(B)\}$ and $N_{+-} = \max\{n_-(A) + n_+(B), n_+(A) + n_-(B)\}$.

A typical case where this result would be useful is when one of the matrices, say $B$, is “almost” definite. For example, if $B$ has many positive eigenvalues, then the inertia of $A$ “almost” determines the number of positive eigenvalues for the pencil, as the corollary below shows.

\textbf{Corollary 1} Let $A, B \in \mathbb{C}^{n \times n}$ be Hermitian and suppose that $B$ has $n - k$ positive eigenvalues. Then the number $n_+(A, B)$ of positive eigenvalues of the pencil $A - zB$ satisfies
\[ n_+(A) - k \leq n_+(A, B) \leq n_+(A) + k. \]

In the talk, I will show analogous bounds for the number zero, negative, nonreal, and infinite eigenvalues of indefinite pencils. Moreover, I will demonstrate that the lower bounds are sharp, in the sense that it is not possible to obtain any better result by using only the information of the inertiae of $A$ and $B$.

Finally, I will discuss extensions of these type of results to general nonlinear eigenvalue problems, as well as applications to the problem of estimating the number of (generalized or nonlinear) eigenvalues lying in a general real interval $(a, b)$. The latter is of interest, for the case of indefinite eigenvalue problems, in several contexts: for instance, the design of efficient eigensolvers based on splitting the spectrum [3], or constraint preconditioning for indefinite linear systems [5].

This talk is based on joint work with Yuji Nakatsukasa, appeared in [7].

\textbf{References}


The SVD and JCF reveal the structure in LOSS OF ORTHOGONALITY

Chris Paige

School of Computer Science, McGill University, Montreal, Quebec, Canada

www.cs.mcgill.ca/~chris/

Abstract

Many numerical algorithms rely on orthogonality, but some which are useful for higher dimensional problems such as Gram-Schmidt orthogonalization, the method of conjugate gradients (CG), the Lanczos process, and the many algorithms based on the Lanczos process and the Golub-Kahan bidiagonalization, can lose significant orthogonality. Fortunately many of these algorithms still produce useful results, and here we show how to derive the structure in any loss of orthogonality in order to analyze any such numerical algorithms.

In dealing with orthogonality and its possible loss we need only consider normalized vectors. Let \( V_k = [v_1, v_2, \ldots, v_k] \in \mathbb{R}^{n \times k} \) with \( \|v_j\|_2 = 1, \ j = 1:k \), then a useful construct for understanding loss of orthogonality in \( V_k \) is \( S_k \), where with strictly upper triangular \( U_k \), see [1], \( (O = \text{orthonormal}) \)

\[
V_k^T V_k = U_k^T + I_k + U_k, \quad (I_k + U_k)(I_k - S_k) = I_k, \quad Q_1 = \begin{bmatrix}
S_k \\
V_k(I_k - S_k)
\end{bmatrix} \in \mathbb{O}^{(k+n) \times k}.
\]

Since \( Q_1^T Q_1 = I_k, \|S_k\|_2 \leq 1 \). Let \( S_k \) have \( m_k \) unit and \( n_k \) zero singular values with SVD

\[
S_k = W \Sigma P^H = W_1 P_1^H + W_2 \Sigma_2 P_2^H, \quad 0 \leq \Sigma = \text{diag}(I_{m_k}, \Sigma_2, O_{n_k}),
\]

\[
P = [P_1, P_2, P_3] \in \mathbb{O}^{k \times k}; \quad W = [W_1, W_2, W_3] \in \mathbb{O}^{k \times k}; \quad P_1, W_1 \in \mathbb{O}^{k \times m_k}; \quad P_3, W_3 \in \mathbb{O}^{k \times n_k};
\]

\[
Q_1 P = \begin{bmatrix}
S_k P \\
V_k(I_k - S_k) P
\end{bmatrix} = \begin{bmatrix}
W_1 \\
W_2 \Sigma_2 \\
V_k(P_1 - W_1) P_2 - W_2 \Sigma_2 P_3
\end{bmatrix} = \begin{bmatrix}
W_1 \\
W_2 \Sigma_2 \\
V_k(P_2 - W_2 \Sigma_2) P_3
\end{bmatrix} = \begin{bmatrix}
W_1 \\
W_2 \Sigma_2 \\
V_k(P_2 - W_2 \Sigma_2) P_3
\end{bmatrix} = \begin{bmatrix}
0 \\
V_k(P_2 - W_2 \Sigma_2) P_3
\end{bmatrix}.
\]

so that \( V_k P_3 \in \mathbb{O}^{n \times n_k} \) shows the orthogonality that is not lost. On the other hand \( V_k(P_1 - W_1) = 0 \) where \( \text{rank}(P_1 - W_1) = m_k \), i.e., the null space of \( V_k \) is \( \text{Range}(P_1 - W_1) \), shows the linear dependence of the columns of \( V_k \). In fact if \( \{1, p, w\} \) is a unit singular triplet (ust) of \( S_k \), i.e., \( S_k p = w, S_k^T w = p, \) and \( \|p\|_2 = \|w\|_2 = 1 \), then it follows from forming \( Q_1 P \) that \( V_k P = V_k w \), revealing this particular rank deficiency of \( V_k \). The converse of this is that if there exist \( y_1 \neq y_2 \) such that \( V_k y_1 = V_k y_2 \), then there exists \( z \) with \( \|z\|_2 = 1 \) such that \( \{1, p = P_1 z, w = W_1 z\} \) is a unit singular triplet of \( S_k \) giving \( V_k P = V_k w \).

The Jordan canonical form (JCF) reveals even finer detail. First we note that all the eigenvalues of \( S_k \) are zero, so that all its right eigenvectors lie in \( \text{Range}(P_3) \), with the left ones in \( \text{Range}(W_3) \). Next if \( \{y_1, y_2, y_3\} \) is a ust of \( S_k \) and \( V_k y_2 = V_k y_1 \) with \( y_1 \in \text{Range}(P_3) \), then defining \( Y_2 = [y_1, y_2] \):

\[
S_k y_2 = y_1, \quad S_k^H y_1 = y_2, \quad S_k y_1 = 0, \quad \|V_k y_2\|_2 = \|V_k y_1\|_2 = 1, \quad S_k Y_2 = Y_2 J_2, \quad J_2 = \begin{bmatrix}
0 & 1 \\
0 & 0
\end{bmatrix}, \quad Y_2^H Y_2 = I_2,
\]

and so \( Y_2 \) is an orthonormal Jordan chain. If we also have \( y_2 \in \text{Range}(W_3) \) then \( S_k Y_2 = Y_2 J_2^T \), and \( Y_2 \) is a complete orthonormal Jordan chain of \( S_k \) of length 2.

For example, in [2] it was shown that the Lanczos process on \( A = A^T \in \mathbb{R}^{n \times n} \) can give a \( V_k \) so that \( S_k \) has several even longer Jordan chains, with, say, \( V_k y_1 = V_k y_2 = \cdots = V_k y_l \) all being good approximations to a single eigenvector of \( A \). This redundancy explains why the computational Lanczos process and the Lanczos-CG method for solution of equations slow down compared with the exact processes. Fortunately the analysis in [2] shows they still give backward stable results.
References


Matrix Equation Methods for Certain Evolutionary Partial Differential Equations

Davide Palitta

Research Group Computational Methods in Systems and Control Theory (CSC), Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstraße 1, 39106 Magdeburg, Germany
http://www.dm.unibo.it/~davide.palitta3/

Abstract

The numerical treatment of partial differential equations (PDEs) often involves a first discretization phase which yields a discrete operator that needs to be inverted. In general, if a \( d \)-dimensional differential operator on a regular domain is discretized with \( n \) nodes in each direction, a common approach consists in writing the discrete problem as a large linear system

\[
Au = f, \quad A \in \mathbb{R}^{n^d \times n^d},
\]

so that well-established procedures, either direct or iterative, can be employed in the solution process. However, in many cases, the coefficient matrix \( A \) in (1) is very structured and a different formulation of the algebraic problem in terms of a matrix equation can be employed. The matrix oriented formulation of the algebraic problems arising from the discretization of certain PDEs is not new. See, e.g., [1, 3, 2]. Nevertheless, only in the last decades the development of efficient solvers for large-scale matrix equations allows for a full exploitation of such reformulation also during the solution phase. See, e.g., [4, 5], and the survey paper [7] for a thorough presentation about solvers for linear matrix equations.

In this talk, we discuss evolutionary PDEs and we show that the aforementioned reformulation in terms of a matrix equation can be performed also for this class of operators. For sake of simplicity, we will restrict our discussion to the heat equation

\[
\begin{align*}
\frac{u_t}{\tau} - \Delta u + f & \quad \text{in } \Omega \times [0,T], \\
u & = g, \quad \text{on } \partial \Omega, \\
u(x,0) & = u_0(x),
\end{align*}
\]

where \( \Omega \subset \mathbb{R}^d \), \( d = 1, 2, 3 \), is a regular domain, but the same strategy can be applied to any PDE of the form \( u_t + L(u) = f \) whenever \( L(u) \) is a linear differential operator involving only space derivatives.

We discretize the problem (2) in both space and time and, if an “all-at-once” approach is considered, the algebraic problem arising from the discretization of (2) amounts to a linear system of the form (1) with \( A \in \mathbb{R}^{n^d \ell \times n^d \ell} \) where \( n \) is the number of nodes employed in each of the \( d \) space directions, \( d \) is the space dimension and \( \ell \) is the number of time steps. As shown in [8], the \( n^d \ell \times n^d \ell \) coefficient matrix \( A \) possesses a Kronecker structure. While in [8] the authors exploit this Kronecker form to design an effective preconditioner for (1), we take advantage of the Kronecker structure to reformulate the algebraic problem in terms of a matrix equation.

For instance, if the backward Euler scheme is employed for the time integration, \( \tilde{\Omega}_h = \{ \tilde{x}_{1d} \} \), \( \tilde{x}_{1d} \in \mathbb{R}^d \), \( \tilde{i}_d = (i_1, \ldots, i_d)^T \in \mathbb{N}^d \), \( i_j = 1, \ldots, n \) for all \( j = 1, \ldots, d \), denotes a uniform discretization of the closed domain \( \tilde{\Omega} \), with \( n \) equidistant points in each of the \( d \) spatial dimensions, and the time interval \([0, T]\) is discretized with \( \ell + 1 \) equidistant nodes \( \{ t_k \}_{k=0,\ldots,\ell} \), then the discretization of (2) leads to

\[
\frac{u_k - u_{k-1}}{\tau} + K_A u_k = f_k, \quad k = 1, \ldots, \ell.
\]
In (3), \(K_d \in \mathbb{R}^{n_d \times n_d}\) denotes the stiffness matrix arising from the discretization of \(-\Delta\) on \(\bar{\Omega}_h\), \(\tau = T/\ell\) is the time-step size, \(f_k \in \mathbb{R}^{n_d}\) collects the space nodal values of \(f\) at time \(t_k\), namely \(f(x_{id}, t_k)\) for all \(x_{id} \in \bar{\Omega}_h\), together with the boundary conditions, while \(u_k\) gathers the approximations to the space nodal values of the solution \(u\) at time \(t_k\), i.e., \(u(x_{id}, t_k)\) for all \(x_{id} \in \bar{\Omega}_h\).1

Rearranging the terms in (3) and applying an all-at-once approach, we get the linear system

\[
\begin{bmatrix}
I_{n_d} + \tau K_d \\
-I_{n_d} & I_{n_d} + \tau K_d \\
& \ddots & \ddots & \ddots \\
& & -I_{n_d} & I_{n_d} + \tau K_d \\
& & & & \ddots & \ddots & \ddots \\
& & & & & 0 & 1 \\
& & & & & & \ddots & 0 \\
& & & & & & & 1 & 0 \\
\end{bmatrix}
\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_\ell \end{bmatrix} =
\begin{bmatrix} u_0 + \tau f_1 \\ \tau f_2 \\ \vdots \\ \tau f_\ell \end{bmatrix},
\quad A \in \mathbb{R}^{n_d \times n_d \times \ell}.
\]

(4)

where \(u_0\) collects the space nodal values of the initial condition \(u_0\).

The coefficient matrix \(A\) in (4) can be written as \(A = I_\ell \otimes (I_{n_d} + \tau K_d) - \Sigma_1 \otimes I_{n_d}\) where

\[
\Sigma_1 = \begin{bmatrix}
0 & 1 & 0 & \cdots & \cdots & \cdots \\
1 & 0 & & & & \\
\vdots & \ddots & \ddots & \ddots & & \\
\cdots & \ddots & \ddots & \ddots & \ddots & \ddots \\
\cdots & \cdots & \cdots & \ddots & \ddots & \ddots \\
1 & \cdots & \cdots & \cdots & \ddots & 0 \\
\end{bmatrix} \in \mathbb{R}^{\ell \times \ell},
\]

and \(\otimes\) denotes the Kronecker product. Therefore, (4) can be reformulated as

\[
(I_{n_d} + \tau K_d)U - U\Sigma_1^T = u_0 e_1^T + \tau [f_1, \ldots, f_\ell],
\quad U = [u_1, \ldots, u_\ell] \in \mathbb{R}^{n_d \times \ell}.
\]

(5)

The matrix equation formulation (5) of the discrete problem naturally encodes the separability of the spatial and time derivatives of the underlying differential operator. We will show that this lets us employ different strategies to deal with the spatial and time components of the algebraic problem and combine them in a very efficient solution procedure. In particular, timely projection techniques are proposed to tackle the spatial operator while the entry-wise structure of the time discrete operator \(\Sigma_1\) is exploited to derive effective solution schemes. The resulting algorithm is able to efficiently solve problems with a tremendous number of degrees of freedom while maintaining a low storage demand as illustrated in several numerical examples.

References


1We assume the entries of both \(f_k\) and \(u_k\) to be sorted following a lexicographic order on the multi-index \(i_d\) for all \(k = 1, \ldots, \ell\).


From a posteriori algebraic error estimator to multilevel iterative solver with $p$-robust behavior

Ani Miraçi, Jan Papeţ, Martin Vohralík

Inria Paris, France & Université Paris-Est, CERMICS (ENPC), France

www.papez.org

Abstract

A posteriori error analysis is an important part of numerical approximation of partial differential equations. In particular, estimation of the discretization error opens door for adaptive techniques where the computational effort is efficiently spent only in some parts of the computational domain. Originally, the estimators were typically derived under the assumption that the associated algebraic system is solved exactly. Starting from pioneering works (see, in particular, [3]), there is, fortunately, a growing body of work avoiding this unrealistic and limiting assumption. Significant research effort is spent in order to derive estimators for the total computational error and the errors of different origin, including the algebraic error, i.e., the error due to inexact algebraic solution, see, e.g., [5, 2] and the references therein.

As highlighted, e.g., in [8], the distribution of the algebraic error can significantly differ throughout the computational domain and can dominate the total computational error in some parts of the domain despite the fact that its (global) norm is smaller than the norm of other sources of error. This is a motivation for developing a posteriori error estimators which allow to estimate the local distribution of errors. This cannot be, in general, provided by a priori bounds. Recently, stopping criteria comparing locally the contribution of errors of different origin were proposed, e.g., in [5, 10]. However, in [5, 10] as well as in many other references, the solution of the (linear) algebraic system and the algebraic error estimation are considered as two separate steps, often involving different computational techniques. When the error indicators suggest that more iterations of the solver are needed to reach a desired tolerance, the work spent in constructing the (local) error indicators is often wasted. In this work, we put forward a construction which links a linear algebraic solver and an algebraic error estimator. Consequently, proving some properties of the error estimator provides us with a proof of some properties of the solver and vice versa.

We consider a conforming finite element discretization of arbitrary polynomial degree $p \geq 1$ of the Poisson model problem. The cornerstone of our estimator and solver definitions lies in the multilevel construction of a residual algebraic lifting, motivated by [9] and done in the spirit of hierarchical stable splittings; see, e.g., [12]. The lifting can be seen as an approximation of the algebraic error, obtained by a V-cycle multigrid method with no pre-smoothing step and a single post-smoothing step. The coarse correction is given by a lowest-order (piecewise affine) function. Our smoothing is chosen in the family of damped additive Schwarz methods applied to overlapping subdomains composed of patches of elements. The additive Schwarz-type smoothing, which can be equivalently written as a block Jacobi smoothing, allows for a parallelizable implementation at each level of the mesh hierarchy. Once this lifting is built, the a posteriori estimator is easily derived as a natural guaranteed lower bound on the algebraic error, following [10]. As our first main result, we show the efficiency of the estimator, i.e., that up to a $p$-robust constant, the estimator is also an upper bound on the error. To the best of our knowledge, an algebraic error estimator with dedicated proof of $p$-robust efficiency has not yet been presented in the literature.

Our solver is then defined as a linear iterative method. Having at hand the residual lifting, which approximates the algebraic error, we use it as a descent direction (the asymmetric approach in
defining the lifting is not a problem for the analysis). The step size is then chosen by a line search in the direction of the lifting. Our choice presents a resemblance with the conjugate gradient method, in that we choose the step size that ensures the best error contraction in the energy norm at the next iteration. As our second main result, we prove that this solver contracts the error at each iteration by a factor bounded independently of $p$. Actually, we also show that the $p$-robust efficiency of the estimator is equivalent to the $p$-robust convergence of the solver.

All these results are derived for a general hierarchy of nested, unstructured, possibly highly refined matching simplicial meshes, and no assumption beyond $u \in H^1_0(\Omega)$ is imposed on the weak solution. The $p$-robustness results rely on the work of [13] for one given mesh. There are also other results on $p$-robustness of a multilevel solver (or preconditioner), mostly for rectangular/hexahedral meshes and discontinuous Galerkin discretizations; see, e.g., [11, 7, 6]. For a recent study on more general meshes, see e.g. Antonietti and Pennesi [1], where a multigrid approach behaves $p$-robustly under the condition that the number of smoothing steps (depending on $p$) is chosen big enough. Similarly to our construction, these results also involve the use of Schwarz-type smoothers, but we recall that we merely need one post-smoothing step. Combined with the results of [9, 10], we then obtain guaranteed and polynomial-degree-robust bounds on all total, algebraic, and discretization error components.

Numerical tests are presented to illustrate our theoretical findings. We also discuss relationships to previous results on stable subspace splittings (decompositions) and associated preconditioners, such as the well-known BPX preconditioner [4].

References


Recent Advances in the Preconditioned Iterative Solution of Time-Dependent PDE-Constrained Optimization Problems

John W. Pearson

University of Edinburgh

https://www.maths.ed.ac.uk/school-of-mathematics/people/person=372

Abstract

Two key applications of numerical linear algebra are in the development of fast and robust solvers for huge-scale matrix systems arising from partial differential equations (PDEs) and optimization problems. In particular, preconditioned iterative methods involve applying accurate and computationally cheap preconditioners at each step of the iterative solver, in order to accelerate its convergence. In this talk, we consider optimization problems where PDEs themselves act as constraints, so-called PDE-constrained optimization. Such problems have numerous applications across science and engineering, for instance in fluid flow control problems, chemical and biological processes, mathematical finance, and medical imaging problems. This talk has a particular focus on devising preconditioners for time-dependent problems.

To give an example of the structure of such a problem, consider the following formulation:

$$\min_{\dot{y}, u} \frac{1}{2} \| y - \hat{y} \|_{Q_1(\Omega)}^2 + \frac{\beta}{2} \| u \|_{Q_2(\Omega)}^2$$

s.t. \( D y = u \) in \( \Omega \),

where \( y \) and \( u \) denote one or more state variables (PDE variables) and control variables respectively, \( \hat{y} \) is a desired state, \( \beta > 0 \) is a regularization parameter, and \( D \) represents some differential operator equipped with boundary and initial conditions. The problem is posed on a space–time domain \( \Omega \), with \( Q_1 \) and \( Q_2 \) denoting two (given) norms. It is possible for additional algebraic constraints to be imposed on the state and/or control variables, such as

\[
y_a \leq y \leq y_b, \quad u_a \leq u \leq u_b \quad \text{a.e. in } \Omega,
\]

for given functions \( y_a, y_b, u_a, u_b \).

For the all-at-once solution of time-dependent problems in particular, applying preconditioned iterative solvers can be particularly advantageous if one is only required to store matrices that have a fraction of the dimension of the matrix system being solved, as for the methods devised in [1] for instance. In order to accomplish this, we consider preconditioners for saddle point systems of the form

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{21} \\ A_{21}^T & A_{22} \\ B_1 & B_2 \\ B_1^T & B_2^T \end{bmatrix} \begin{bmatrix} y \\ u \\ p \end{bmatrix} = b,$$

where the individual blocks of the matrix system are highly structured, and are frequently sparse. In particular, we devise effective approximations of the \((1,1)\)-block \( A \) and (negative) Schur complement \( S := BA^{-1}B^T \), and use these approximations within block preconditioners. Here \( y, u, \) and \( p \) denote the discretized versions of the state, control, and adjoint variables respectively. For a number of such problems, it is possible to devise solvers which are robust with respect to all parameters within the problem statement, and if sparse matrices are involved the solvers can be such that the computational costs are linear with respect to the dimension of the system.
In this talk, we discuss new directions in the construction of iterative solvers for time-dependent PDE-constrained optimization problems. Our focus is on novel solution strategies and insights for three specific classes of problems:

- Preconditioned iterative solvers for problems with additional algebraic constraints, along with sparsity-promoting terms within the cost functional, solved using an interior point method (IPM) [2]. Here we extend the IPM–Krylov solvers devised in [3], to problems where the norm $Q_2$ involves an $L^1$-norm term in order to promote sparsity within the solution for the control variable. Strategies for smoothing the $L^1$ norm are coupled with new preconditioners for the resulting matrix systems.

- Iterative solution of optimization problems constrained by certain space–time fractional differential equations (FDEs) [4]. For such problems, the resulting matrix systems are not only of huge scale, but the matrices relating to the FDE constraints are dense. We are able to utilize the multilevel Toeplitz structure of many of the blocks of the matrix, in order to derive multilevel circulant preconditioners that lead to very reasonable storage requirements and computational operation costs.

- Deferred correction techniques for increasing the order of accuracy in the time variable for PDE-constrained optimization problems [5]. It is possible to construct “outer solves” for the coupled systems of PDEs, where at each iteration we approximately solve a matrix system for a measure of the errors obtained at the previous deferred correction step. Crucially, it is possible to apply existing preconditioners which have previously been devised for problems solved with lower-order time-stepping schemes, and utilize the deferred correction technique to greatly reduce the discretization error in the time variable.

We also aim to provide an outlook into the subject area, in particular by discussing further modern applications which could be tackled using the methodology presented in this talk.

References


Preconditioners for Symmetrized Toeplitz and Multilevel Toeplitz Problems

Jennifer Pestana

Department of Mathematics and Statistics, University of Strathclyde
http://personal.strath.ac.uk/jennifer.pestana/

Abstract

Linear systems \( Ax = b \), with \( A \) a real \( n \times n \) Toeplitz or multilevel Toeplitz matrix, arise in numerous applications. These include differential and integral equations discretized on uniform meshes, time series analysis, and signal and image processing. Latterly, interest has also been driven by the need for fast solvers for problems involving fractional derivatives, which are highly nonlocal.

For ease of exposition, we consider here only Toeplitz matrices

\[
A = \begin{bmatrix}
a_0 & a_{-1} & \ldots & a_{-n+2} & a_{-n+1} \\
a_1 & a_0 & a_{-1} & \ldots & a_{-n+2} \\
\vdots & a_1 & a_0 & \ddots & \vdots \\
a_{n-2} & \ddots & \ddots & \ddots & a_{-1} \\
a_{n-1} & a_{n-2} & \ldots & a_1 & a_0
\end{bmatrix},
\]

although we stress that our results extend straightforwardly to multilevel problems. We additionally assume that \( A \) is generated by a symbol \( f \in L^1([−π, π]) \) in the sense that

\[
a_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\theta)e^{-ik\theta} d\theta, \quad k \in \mathbb{Z},
\]

and use the notation \( A(f) \) to emphasize this link; this assumption is satisfied in many applications [1, p. 13]. Properties of \( A(f) \) are then naturally related to those of \( f \). For example, if \( f \) is real-valued then \( A(f) \) is Hermitian and its eigenvalues are characterized by \( f \); if \( f \) is also essentially positive then \( A(f) \) is Hermitian positive definite.

Since the groundbreaking work of Olkin [5] and Strang [9], it has become commonplace to solve these Toeplitz systems by Krylov subspace methods. Early (circulant) preconditioners were constructed from the entries of the coefficient matrix, but now many preconditioners are based on \( f \), since the symbol can profoundly influence properties of \( A(f) \) and hence the convergence rate of the Krylov method [1]. When \( A(f) \) is symmetric, MINRES or CG convergence theory guides the choice of preconditioner and allows for guaranteed upper bounds on convergence rates. However, for nonsymmetric \( A(f) \) this is not possible in general because it is difficult to accurately capture the effect of nonnormality on the convergence rates of nonsymmetric Krylov solvers.

Fortuitously, if \( Y \in \mathbb{R}^{n \times n} \), \( y_{jk} = \delta_{j+k-n-1} \) is the reverse identity matrix then \( YA(f) \) is Hankel, hence symmetric. Consequently, \( YA(f)x = Yb \) can be solved by the preconditioned MINRES method with its short-term recurrences, optimality property and descriptive convergence bounds, as in [7]. However, the “preconditioner” \( Y \) does not, in general, accelerate convergence; indeed roughly half the eigenvalues of \( YA(f) \) are distributed like a uniform sampling of \( |f| \) over \([−π, π]\), while the other half are distributed like a uniform sampling of \(-|f|\) over the same interval [2, 4]. For many practical problems this results in unfavourable spectral properties and unacceptably slow MINRES convergence rates. Instead, the role of \( Y \) is to enable us to use MINRES, with its desirable properties, and design a secondary preconditioner \( P \) that is guaranteed to accelerate convergence.
Since \( P \) must be symmetric positive definite, but \( A(f) \) is nonsymmetric, preconditioning the symmetrized problem has proved challenging. Certain absolute-value circulants can be applied, as suggested in [7], but their performance is not always satisfactory, and multilevel versions are not optimal [8]. More fundamentally, since the symbol \( f \) is so closely related to properties of \( Y A(f) \), we would like preconditioners to be based in some way on this function. Here, we describe some preconditioners of this type, that were recently introduced in [6].

If \( f = f_R + i f_I \), with \( f_R, f_I \) real-valued functions, then a candidate preconditioner is the Toeplitz matrix \( A(f_R) \), provided that \( f_R \) is essentially positive (so that \( A(f_R) \) is symmetric positive definite). Furthermore, the eigenvalues of \((A(f_R))^{-1}(Y A(f)) \) lie in \([-1 - \epsilon, -1] \cup [1, 1 + \epsilon]\), where

\[
\epsilon < \text{esssup}_{\theta \in [-\pi, \pi]} \left| \frac{f_I(\theta)}{f_R(\theta)} \right|.
\]

This shows that the eigenvalues are uniformly bounded away from zero, and that the spectrum is nicely distributed when our original coefficient matrix \( A(f) \) is close to symmetric, as might be expected. Since \( A(f_R) \) corresponds to the symmetric part of \( A(f) \), this matrix is trivial to compute and, for certain problems, systems with \( A(f_R) \) are easy to solve. If this is not the case, we show how to construct, e.g., circulant, banded Toeplitz or multigrid approximations that lead to cheap and effective preconditioners for many problems.

If \( A(f) \) is far from symmetric a more suitable preconditioner may be obtained by considering the Toeplitz matrix \( A(|f|) \). The eigenvalues of \((A(|f|))^{-1}(Y A(f)) \) lie in \([-1 - \epsilon, 1 + \epsilon]\), where \( \epsilon = o(n) \) as \( n \to \infty \), and for large enough problems most eigenvalues cluster at \(-1\) and \(1\) [6]. Although \( A(|f|) \) is typically dense it can again be approximated, e.g., by a circulant or banded Toeplitz matrix, and our numerical experiments demonstrate the effectiveness of these approximations.

Although these results hold only for (multilevel) Toeplitz matrices, we will also discuss more recent work on Krylov subspace methods for certain “Toeplitz-like” problems. These generalized locally Toeplitz (GLT) matrices [3] are obtained by, e.g., discretizing differential equations with variable coefficients or using non-uniform meshes. It has been shown that many properties of a GLT matrix are also closely related to a scalar-valued function, that can again be used to develop effective preconditioners. Here we discuss the implications for preconditioners and Krylov solvers.

References


Recent development of the core problem theory in the context of the total least squares minimization

Iveta Hnětynková, Martin Plešinger, and Jana Žáková

Department of Mathematics and Didactics of Mathematics, Technical University of Liberec
www.fp.tul.cz/~plesinger

Abstract

We consider an orthogonally invariant linear approximation problem

\[ A(x) \approx b, \quad \text{where} \quad A \in L(U, V), \quad x \in U, \quad b \in V \setminus \text{Im}(A), \]

and \( U, V, L(U, V) \) are finite-dimensional normed vectors spaces. The incompatibility in the problem is caused by errors contained in the right-hand side \( b \) as well as in the mapping \( A \). By the total least squares (TLS) we understand replacing the original problem by a minimization

\[
\min_{E \in L^*, g \in V} (\|E\|^2 + \|g\|^2)^{\frac{1}{2}} \quad \text{subject to} \quad (b + g) \in \text{Im}(A + E),
\]

where \( L^* \) is some given sub-set (e.g., a sub-space or sub-manifold) of \( L(U, V) \).

TLS minimization has been studied for a long time in different specific settings. The early groundwork by Golub and Van Loan [1] focuses on the single (or vector) right-hand side case

\[ Ax \approx b, \quad \text{where} \quad A \in \mathbb{R}^{m \times n}, \quad x \in \mathbb{R}^n, \quad b \in \mathbb{R}^m. \]

It was afterwards complemented, in particular, by works of Van Huffel and Vandewalle [9], and Wei [10, 11], focused on the multiple (or matrix) right-hand side case

\[ AX \approx B, \quad \text{where} \quad A \in \mathbb{R}^{m \times n}, \quad X \in \mathbb{R}^{n \times d}, \quad B \in \mathbb{R}^{m \times d}. \]

There is a principal difficulty in the TLS minimization, that already the simplest (vector right-hand side) formulation may not have a solution; see [1]. The non-existence of a TLS solution motivates, among others, the so-called non-generic approach; see [9].

The core problem concept, introduced by Paige and Strakoš in [8], brings an important insight into the vector right-hand side case. The concept is based on existence of the orthogonal transformation

\[
P^T[b, A] = \begin{bmatrix} 1 & 0 \\ 0 & Q \end{bmatrix} = \begin{bmatrix} b_1 & A_{11} & 0 \\ 0 & 0 & A_{22} \end{bmatrix}, \quad P^T = P^{-1}, \quad Q^T = Q^{-1},
\]

that splits the original problem \( Ax \approx b \) into two independent sub-problems: The so-called core problem \( A_{11}x_1 \approx b_1 \), which we are interested in, and a homogeneous problem \( A_{22}x_2 \approx 0 \). The core problem always has the unique TLS solution and contains all the necessary and only the sufficient information for solving the original problem; the other problem contains all the redundancies and irrelevant information; see [8]. While separating the useful and relevant information from the rest, the transformation also allows to answer two important questions: ‘Why and when the TLS solution does not exist?’, and ‘What is the meaning of the non-generic solution?’; see [8].

The generalization of the core problem concept to the matrix right-hand side case has been introduced in a series of papers [3, 4, 2]. The analysis presented there reveals various difficulties. In particular, the matrix right-hand side core problem still may not have a TLS solution. In our contribution we would like to present our recent results in the analysis and extensions of the core problem concept:
First we look *inside* the matrix right-hand side core problem \( A_{11}X_{11} \approx B_1 \) and study its possible internal structure. The core problem revealing transformation is here followed by another orthogonal transformation possibly decomposing the core problem into several smaller sub-problems. Schematically:

\[
[B, A] \mapsto \begin{bmatrix} B_1 & 0 & A_{11} & 0 \\ 0 & 0 & 0 & A_{22} \end{bmatrix}; \quad \text{then} \quad [B_1, A_{11}] \mapsto \begin{bmatrix} B_{1,\alpha} & 0 & A_{11,\alpha} & 0 \\ 0 & B_{1,\beta} & 0 & A_{11,\beta} \end{bmatrix}.
\]

This (de)composition and its behavior with respect to sub-problems with various properties is analyzed in [2] and [7].

Then we look in the *surroundings* of the matrix right-hand side core problem. We extend the analysis to several closely related formulations. The first straightforward generalization of the vector and matrix problem is a \((k\text{-way}, k > 2)\) tensor right-hand side problem

\[ A \times_1 X \approx B, \quad \text{where} \quad A \in \mathbb{R}^{m \times n}, \quad X \in \mathbb{R}^{n \times d_1 \times \ldots \times d_k}, \quad B \in \mathbb{R}^{m \times d_2 \times \ldots \times d_k}. \]

Another generalization is based on a simple observation that \( A \in \mathbb{R}^{m \times n} \) can represent mappings only from a very specific proper sub-space \( \mathcal{L}_* \) of \( \mathcal{L}(\mathbb{R}^{n \times d}, \mathbb{R}^{m \times d}) \). By considering, e.g.,

\[ A_L X A_R^T \approx B, \quad \text{where} \quad A_L \in \mathbb{R}^{m \times n}, \quad A_R \in \mathbb{R}^{c \times d}, \quad X \in \mathbb{R}^{n \times d}, \quad B \in \mathbb{R}^{m \times c}, \]

the set \( \mathcal{L}_* \) of allowed corrections is substantially enlarged. Extensions of the core problem theory to these cases are presented in [5] and [6], respectively. There we have proved the existence of core problems and described their basic properties. Relations between the individual formulations from the TLS point of view, are under investigation.

### References


Solving Singular Generalized Eigenvalue Problems by Rank-Completing Perturbations

Michiel E. Hochstenbach, Christian Mehl, and Bor Plestenjak

IMFM and Faculty of Mathematics and Physics, University of Ljubljana, Slovenia
https://www.fmf.uni-lj.si/~plestenjak/

Abstract

Singular generalized eigenvalue problems are very challenging to solve, both with respect to accuracy and efficiency. We propose a simple method based on one perturbation of the original problem.

Let \( A - \lambda B \) be an \( n \times n \) singular pencil, i.e., \( \det(A - \lambda B) \equiv 0 \). Then \( \lambda_0 \) is a finite eigenvalue of \( A - \lambda B \) if \( \text{rank}(A - \lambda_0 B) < \text{nrank}(A, B) \), where \( \text{nrank}(A, B) := \max_{\zeta \in \mathbb{C}} \text{rank}(A - \zeta B) \) is the normal rank of \( A - \lambda B \). Similarly, \( \infty \) is an eigenvalue if \( \text{rank}(B) < \text{nrank}(A, B) \). We only consider square pencils as rectangular case can be reduced to the square case by adding zero rows or columns.

Let \( \text{nrank}(A, B) = n - k \). We will show that the eigenvalues of \( A - \lambda B \) can be efficiently computed with the help of just one rank-completing perturbation of the form

\[
\tilde{A} - \lambda \tilde{B} := A - \lambda B + \tau (UD_A V^* - \lambda U D_B V^*),
\]

where \( U, V \) are \( n \times k \) matrices with orthonormal columns, \( D_A, D_B \) are diagonal \( k \times k \) matrices, and \( \tau \neq 0 \). The eigenvalues of the original pencil are the eigenvalues of the perturbed regular pencil such that the corresponding left and right eigenvectors are orthogonal to \( U \) and \( V \), respectively.

Theorem 1 [4] Let \( A - \lambda B \) be an \( n \times n \) singular pencil of normal rank \( n - k \) with \( r \) eigenvalues, let \( U, V \in \mathbb{C}^{n,k} \) have full column rank and let \( D_A, D_B \in \mathbb{C}^{k,k} \) be such that \( D_A - \lambda D_B \) is regular and all eigenvalues of \( D_A - \lambda D_B \) are distinct from the eigenvalues of \( A - \lambda B \). Then, for generic \( U \) and \( V \), we can classify all eigenvalues of the perturbed pencil (1) into the following three groups:

1. True eigenvalues: There are \( r \) such eigenvalues that are exactly the eigenvalues of \( A - \lambda B \). The matching right eigenvectors \( x \) and left eigenvectors \( y \) satisfy \( V^* x = 0 \) and \( U^* y = 0 \).

2. Prescribed eigenvalues: There are \( k \) such eigenvalues that are equal to the eigenvalues of \( D_A - \lambda D_B \). The matching right and left eigenvectors \( x \) and \( y \) satisfy \( V^* x \neq 0 \) and \( U^* y \neq 0 \).

3. Random eigenvalues: The remaining \( n - r - k \) eigenvalues are simple and if \( x \) and \( y \) are the matching right and left eigenvector of such an eigenvalue, then either \( V^* x = 0 \) and \( U^* y \neq 0 \), or \( V^* x \neq 0 \) and \( U^* y = 0 \).

Singular generalized eigenvalue problems are ill-posed as arbitrarily small perturbations may cause drastic changes in the eigenvalues. A striking example from [5] is

\[
A - \lambda B = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \tilde{A} - \lambda \tilde{B} = \begin{bmatrix} 1 & \varepsilon_1 \\ \varepsilon_2 & 0 \end{bmatrix} - \lambda \begin{bmatrix} 1 & \varepsilon_3 \\ \varepsilon_4 & 0 \end{bmatrix},
\]

where \( \varepsilon_1, \ldots, \varepsilon_4 \neq 0 \). While \( A - \lambda B \) is singular and has one eigenvalue 1, \( \tilde{A} - \lambda \tilde{B} \) is regular and has two eigenvalues \( \frac{1}{\varepsilon_3} \) and \( \frac{1}{\varepsilon_4} \) that can be anywhere in the complex plane even for tiny \( \varepsilon_1, \ldots, \varepsilon_4 \).

As observed in [9] a situation as above is exceptional and generically small perturbations of a singular pencil make the pencil regular and some of the eigenvalues of the perturbed pencil are very close to the original eigenvalues of the singular pencil; an explanation is provided in [2].
A usual approach for a singular generalized eigenvalue problem is to first extract the regular part and then apply a standard eigenvalue solver, e.g., the QZ algorithm, to it. An example is GUPTRI [1], which applies a staircase algorithm in the first step. GUPTRI is potentially slow as the worst case complexity of the staircase algorithm is $O(n^4)$. In addition, staircase algorithms can fail as they rely on rank decisions that can be difficult to make due to roundoff errors, see, e.g., [3, 7].

A different class of algorithms is based on the observation that random small perturbations move the true eigenvalues only slightly. Such algorithm was first proposed in [6], where a singular pencil is perturbed twice and the obtained eigenvalues are compared with each other in order to find the true eigenvalues. Our algorithm, which is based on Theorem 1, belongs to this group, but requires only one perturbation.

In addition, we can use $s(\lambda_i) = y_i^* \tilde{B} x_i$, where $\lambda_i$ is an eigenvalue of (1) and $y_i$ and $x_i$ are the corresponding left and right eigenvector, to further separate true eigenvalues into finite and infinite ones. Namely, for a simple finite true eigenvalue $s(\lambda_i) \neq 0$, while $s(\lambda_i) = 0$ for an infinite true eigenvalue. Based on this and Theorem 1, we can extract finite eigenvalues of $A - \lambda B$ using the following algorithm. For numerical examples and implementation, see [4] and [8].

Algorithm 1: Computing finite eigenvalues of a singular pencil $(A, B)$.

**Input:** $A$ and $B$, where $\|A\|_1 \approx \|B\|_1 \approx 1$, constant $\tau (10^{-2})$, thresholds $\delta_1 (10^{-8})$ and $\delta_2 (10^{-14})$.

**Output:** Eigenvalues of the finite regular part.

1. Compute $\text{null rank}(A, B): k = n - \text{null rank}(A - \zeta B)$ for a random $\zeta$.
2. Select random $n \times k$ matrices $U$ and $V$ with orthonormal columns.
3. Select diagonal $k \times k$ matrices $D_A$ and $D_B$.
4. Compute the eigenvalues $\lambda_i$, $i = 1, \ldots, n$, and right and left eigenvectors $x_i$ and $y_i$ of (1).
5. Compute $s_i = y_i^* \tilde{B} x_i$ and $\zeta_i = \max(\|V^* x_i\|, \|U^* y_i\|)$ for $i = 1, \ldots, n$.
6. Return all eigenvalues $\lambda_i$, $i = 1, \ldots, n$, where $\zeta_i < \delta_1$ and $|s_i| > \delta_2$.

**References**


Balanced Truncation for Linear System with Quadratic Output: Theory, Error Bounds and Numerics

Peter Benner, Pawan Goyal, and Igor Pontes Duff

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany

http://ipontes.github.io

Abstract

Balanced truncation is one of the most common model order reduction techniques. This method mainly relies on reachability and observability energy functionals. For linear systems, these functionals are encoded by the reachability and observability Gramians [2]. In this talk, we discuss an extension of balanced truncation for a variant of LTI systems, namely linear system with quadratic output (LTI_QO), whose dynamics are governed by

\[
\Sigma := \begin{cases} 
\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = 0, \\
y(t) = x(t)^T M x(t),
\end{cases}
\]

where \(x(t) \in \mathbb{R}^n\), \(A, M \in \mathbb{R}^{n \times n}\) and \(B \in \mathbb{R}^{n \times m}\). Due to the quadratic term in the output equation, this system represents a nonlinear input-output mapping. This class of systems appears, particularly, when one’s interests lie in observing, e.g., the variance or standard deviation of the state variables from a reference point. This happens to be the case in random vibration analysis and problems where response quantities related to energy or power are considered. Additionally, a quadratic observation also appears in electrical circuits combined with time-harmonic Maxwell’s equations.

MOR techniques for LTI_QO systems have been investigated in the past. For instance, the authors in [3] have proposed to rewrite an LTI_QO system as an LTI system. It is then followed by reducing with well-known techniques for LTI systems such as balanced truncation and interpolation-based methods. Furthermore, very recently, the authors in [4] proposed an alternative approach, where the LTI_QO system is written as a quadratic-bilinear (QB) system. Subsequently, the QB system is reduced by tailoring the balanced truncation approach, proposed in [6]. However, in the former approaches, one does not directly utilize the quadratic structure of the output equation, and the latter approach is not only numerically expensive, but it also fails to keep the structure of the original system in the reduced systems.

In the present work, we design a balanced truncation procedure for LTI_QO systems enabling us to preserve the quadratic output structure in the reduced-order system. To that end, we propose a new algebraic observability Gramian \(Q\) for the system (1) based on Hilbert space adjoint theory [5]. We then show that it satisfies a certain type of generalized Lyapunov equation, i.e.,

\[ A^T Q + AQ + MPM = 0, \]

where \(P\) is the controllability Gramian which satisfies the following standard Lyapunov equation

\[ AP + PA^T + BB^T = 0. \]

Furthermore, we investigate the connection of the newly proposed observability Gramian to an observability energy functional. This allows us to find those states that are hard to control and hard
to observe via an appropriate balancing transformation. Truncating such states yields reduced-order systems. Additionally, we show that the proposed methodology preserves the stability. Moreover, the solutions of the above Lyapunov equations often exhibit low-rank phenomena, i.e., there exist
$Z_Q \in \mathbb{R}^{n \times l_Q}$ and $Z_P \in \mathbb{R}^{n \times l_P}$, with $l_Q, l_P \ll n$, such that $Q \approx Z_Q Z_Q^T$ and $P \approx Z_P Z_P^T$. Thus, we make use of low-rank solvers of Lyapunov equations, which are not only numerically efficient but also yield the Gramians directly in low-rank factor form, which is desirable for constructing reduced-order models.

Finally, we propose the concept of $\mathcal{H}_2$ norm and inner product for LTI $QO$ systems. As a consequence, we are able to derive a priori error bounds as follows:

$$\|y - \hat{y}\|_{L_\infty} \leq \|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_2} \|u \otimes u\|_{L_2}.$$  

The above inequality suggests that if the reduced-order model is close enough to the original one with respect to the $\mathcal{H}_2$ norm, they both will produce close outputs. Moreover, we show that $\|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_2}$ depends on the neglected generalized Hankel singular values.

The efficiency of the proposed method is demonstrated by means of various semi-discretized non-linear partial differential equations and compared with the other model reduction techniques from the literature, e.g., [3, 4].

The presented results are available in [1].

References


Best $k$-layer neural network approximations

Lek-Heng Lim, Mateusz Michalek, and Yang Qi

Department of Mathematics, University of Chicago
math.uchicago.edu/~yangqi/

Abstract

The training of a neural network may essentially be viewed as a matrix approximation problem. Given a training set $x_1, \ldots, x_n \in \mathbb{R}^{n_1}$ with corresponding responses $y_1, \ldots, y_n \in \mathbb{R}^{n_{k+1}}$, fitting a $k$-layer neural network requires that one estimates the weights, that is, $k$ matrices and $k$ vectors $\theta := (A_1, b_1, A_2, b_2, \ldots, A_k, b_k)$ in

$$
(\mathbb{R}^{n_2 \times n_1} \times \mathbb{R}^{n_2}) \times (\mathbb{R}^{n_3 \times n_2} \times \mathbb{R}^{n_3}) \times \cdots \times (\mathbb{R}^{n_{k+1} \times n_k} \times \mathbb{R}^{n_{k+1}}) =: \Theta,
$$

via the following best $k$-layer neural network approximation problem:

$$
\inf_{\theta \in \Theta} \sum_{i=1}^{n} \| y_i - (A_k \sigma(\cdots (A_2 \sigma(A_1 x_i - b_1) - b_2) \cdots) - b_k) \|^2_2.
$$

The function $\sigma$ is called the activation and is applied coordinatewise to a vector. Common choices include $\sigma(x) = \max(x, 0)$, $\sigma(x) = \frac{1}{1+\exp(-x)}$, or $\sigma(x) = \tanh(x)$. If we write $\nu_{\theta}$ for the neural network with weights $\theta$, then (2) may be written in the simple form

$$
\inf_{\theta \in \Theta} \| Y - \nu_{\theta}(X) \|_F,
$$

where $X = [x_1, \ldots, x_n] \in \mathbb{R}^{n_1 \times n}$ and $Y = [y_1, \ldots, y_n] \in \mathbb{R}^{n_{k+1} \times n}$. In machine learning parlance, performing (2) or (3) is known as empirical risk minimization.

The appearance of the nonlinear $\sigma$ in (2) is the point of departure from existing matrix approximation problems (e.g., total least squares) in numerical linear algebra. Nevertheless we have shown in [1] that a best $k$-layer neural network approximation bears many similarities to the best rank-$r$ approximation and other related problems that we studied in [2].

The most important revelation is that (2) has no solution in general. We show in [1] that even for $k = 2$, the infimum

$$
\inf_{A,C \in \mathbb{R}^{n \times n}; b,d \in \mathbb{R}^{n}} \sum_{i=1}^{n} \| y_i - (C \sigma(A x_i - b) - d) \|^2_2
$$

is not attainable in general for common activations $\sigma$ like ReLU, sigmoid, or hyperbolic tangent. In addition, we show that for smooth activations, such failure to attain an infimum can happen on a positive-measured subset of responses. The phenomenon reminds one of the nonexistence of best rank-$r$ approximations for higher-order tensors [2]. The explanation is also similar — just as the set of tensors of rank not more than $r$ is not a closed set in general, the image of weights of a $k$-layer neural network, i.e.,

$$
\{ \nu_{\theta}(X) \in \mathbb{R}^{n_{k+1} \times n} : \theta \in \Theta \}
$$

is also not a closed set in general, and thus we do not expect an optimum $\theta$ in (3) to be attainable. The geometry involved for demonstrating the nonexistence of a solution to (3) is however more subtle than that for the best rank-$r$ approximation problem for tensors — for example, with the ReLU activation $\sigma(x) = \max(0, x)$ and $k = 2$, the set in (5) has the structure of a join locus of a secant locus and a line. A careful study of this geometry allows us to completely classify the cases where the approximation problem (4) for a two-layer ReLU-activated neural network attains its infimum.
References


Novel techniques for exploiting low-rank structures in nonlinear eigenvalue problems

Javier Pérez and María C. Quintana

Departamento de Matemáticas, Universidad Carlos III de Madrid, Spain.
https://matematicas.uc3m.es/index.php/maquinta-menu

Abstract

Given a nonlinear matrix-valued function $F : \mathbb{C} \rightarrow \mathbb{C}^{m \times m}$, the nonlinear eigenvalue problem (NLEP) consists of computing scalars $\lambda \in \mathbb{C}$ and non-zero vectors $v \in \mathbb{C}^m$ such that

$$F(\lambda)v = 0,$$

under the regularity assumption $\det(F(\lambda)) \neq 0$. NLEPs arise in a variety of applications in physics and engineering [4].

A popular method for solving NLEPs is based on polynomial or rational approximation [5, 8, 9]. This approach transforms a NLEP into a polynomial or rational eigenvalue problem (PEP or REP, respectively). The resulting PEP or REP can be solved by linearization, which replaces the PEP or the REP with a generalized eigenvalue problem preserving the spectral information.

Any nonlinear matrix-valued function $F(\lambda)$ can always be written in the form

$$F(\lambda) = A_0f_0(\lambda) + A_1f_1(\lambda) + \cdots + A_nf_n(\lambda),$$

for some non-linear scalar functions $f_i(\lambda)$ and constant matrices $A_i$. In practice, some (many) of the coefficients $A_i$ have low rank, a property that is inherited by the polynomial or rational approximants. In recent years, the importance of exploiting low-rank structures in the numerical solution of NLEPs has become increasingly well-recognized. This has a number of relevant benefits. For example, it leads to smaller linearized eigenvalue problems and to more efficient Krylov methods in the large-scale setting.

In this talk, using the theory of polynomial system matrices of rational transfer functions and the new definition of linearization in [1], we present a variety of techniques for exploiting the low-rank property of the matrix coefficients of a nonlinear eigenvalue problem. We show how to build trimmed linearizations that allow us to recover all the spectral information of the original polynomial or rational eigenvalue problem. In addition, we show how to build linearizations for matrix polynomials that automatically deflate eigenvalues at zero or/and at infinity without any computational cost. This new deflation technique is much simpler than other approaches based on combining rank-revealing decompositions with equivalence transformations [2, 3, 6, 7]. The advantages of our low-rank-exploitation techniques will be illustrated with numerical examples.

References


Krylov meets Bregman

Alessandro Buccini, Mirjeta Pasha, and Lothar Reichel

Department of Mathematical Sciences, Kent State University

www.math.kent.edu/~reichel

Abstract

Many problems in Science and Engineering require the computation of a solution that is sparse when expressed as a linear combination of suitable functions or vectors. Being sparse means that many coefficients in the linear combination vanish. Problems of this kind arise, e.g., in image restoration and micro-array analysis.

We are concerned with the computation of a sparse solution of problems of the form

$$\min_{u \in \mathbb{R}^n} \|Au - b\|_2,$$

where $A \in \mathbb{R}^{m \times n}$, with $m \leq n$, is a large matrix, whose singular values gradually decay to zero with no significant gap; the matrix is very ill-conditioned and may be rank deficient. The vector $b \in \mathbb{R}^m$ is determined by data that is corrupted by an unknown error $e \in \mathbb{R}^m$, which typically stems from measurement inaccuracies. Problems of this kind arise, for instance, in image deblurring, and are commonly referred to as linear discrete ill-posed problems.

Let $b_{\text{true}}$ denote the unknown error-free vector associated with $b$, i.e., $b = b_{\text{true}} + e$, and let $A^\dagger$ stand for the Moore–Penrose pseudo-inverse of $A$. We would like to determine an accurate approximation of $u_{\text{true}} = A^\dagger b_{\text{true}}$, where we assume that $u_{\text{true}}$ is known to be sparse. Straightforward solution of (1) typically does not give a useful approximation of $u_{\text{true}}$ due to the error $e$ in $b$ and the ill-conditioning of $A$.

Assume that the matrix $A$ is surjective. An approach to reducing the propagation of the error $e$ into the computed solution when $m \ll n$ is to seek a sparse approximate solution of (1). Such an approximate solution can be computed by solving the constrained minimization problem

$$u_\mu = \arg \min_{u \in \mathbb{R}^n} \left\{ \mu \|u\|_1 + \frac{1}{2\delta} \|u\|_2^2 : Au = b \right\},$$

which has a unique solution. Here $\mu > 0$ and $0 < \delta < 1/\rho(A^T A)$ are user-defined constants, with $\rho(A^T A)$ denoting the spectral radius of $A^T A$. We refer to $\mu > 0$ in (2) as the regularization parameter. The upper bound for $\delta$ secures convergence of the iterates generated by the linearized Bregman algorithm (LBA). This algorithm is a popular iterative method for the solution of (2); see, e.g., [1, 2] and references therein. It is defined as follows: Let $T_\mu(v)$ denote the soft-thresholding operator, i.e.,

$$T_\mu(v) = [t_\mu(v_1), t_\mu(v_2), \ldots, t_\mu(v_n)]^T \in \mathbb{R}^n \quad \text{for} \quad v = [v_1, v_2, \ldots, v_n]^T,$$

where

$$t_\mu(x) = \begin{cases} 
0 & \text{if } |x| \leq \mu, \\
\text{sign}(x)(|x| - \mu) & \text{if } |x| > \mu.
\end{cases}$$

The LBA iterations with $u^0 = v^0 = 0$ can be written as, for $k = 0, 1, 2, \ldots$,

$$v^{k+1} = v^k - A^T(Au^k - b),$$

$$u^{k+1} = \delta T_\mu(v^{k+1}).$$
The vectors $u^{k+1}_k, k = 1, 2, \ldots$ , are approximate solutions of (2). The iterations are terminated when two consecutive iterates $u^k$ are sufficiently close.

Images usually have sparse representations in terms of wavelets or framelets. The exploitation of this property often increases the quality of the computed solution in image restoration problems. To make use of the sparsity, we transform the problem (2) so that its solution has a sparse representation.

The iterates determined by the LBA may converge very slowly when $A$ is ill-conditioned. We are interested in speeding up the convergence. Numerical experiments indicate that restricting the iterates generated by the LBA to a Krylov subspace generated by a few steps of Golub–Kahan bidiagonalization applied to $A$ speeds up the convergence significantly. We will describe such a Krylov-Bregman iterative method, discuss how many steps of Golub–Kahan bidiagonalization to carry out, and analyze its convergence properties. We refer to this algorithm as the Krylov LBA. Table 1 provides an example. The problem solved is an image deblurring problem. The image restored is of a telescope and is represented by $986 \times 986$ pixels. The PSF, which defines the matrix $A \in \mathbb{R}^{986 \times 986}$ models Gaussian blur. The noise level is the relative error in $b$.

Table 1: Comparison of the LBA and Krylov LBA in terms of RRE (relative restoration error), number of iterations, and CPU time for two noise levels.

<table>
<thead>
<tr>
<th></th>
<th>LBA</th>
<th>Krylov LBA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RRE</td>
<td>Number of iterations</td>
</tr>
<tr>
<td>Telescope 1%</td>
<td>0.0899</td>
<td>145</td>
</tr>
<tr>
<td>Telescope 5%</td>
<td>0.1316</td>
<td>27</td>
</tr>
</tbody>
</table>

Table 1 shows the Krylov LBA to determine image restorations with somewhat smaller RRE in much less CPU time than the LBA. A standard laptop computer was used for the computations.

In many applications it is known that the desired solution $u_{\text{true}}$ lies in a closed and convex set. In this situation, it is generally beneficial to impose constraints on the iterates generated by LBA and Krylov LBA so that they lie in this closed and convex set. For instance, in image restoration problems, the entries of the desired solution represent nonnegative pixel values of an image. It is generally meaningful to solve the constrained minimization problem

$$u^*_\mu = \arg \min_{u \geq 0} \{ \mu \| u \|_1 + \frac{1}{2\delta} \| u \|_2^2 : Au = b \}$$

instead of (2). We will analyze the convergence of the Krylov-Bregman iteration with convex constraints. Computed examples will be presented.

If time permits, then we also will discuss Krylov-FISTA methods.

References

Improving Krylov subspace approximations for solution of ill-posed inverse problems: oversampling and truncation

Anthony Helmstetter, Rosemary Renaut

School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ, USA
https://math.la.asu.edu/~rosie/

Abstract

The LSQR iterative method has long been a standard for the solution of large-scale ill-posed linear problems, [6]. The Krylov space used for the solution of the problem yields a surrogate space, much smaller than the original space, that inherits the ill-conditioning of the original problem. More recently, the randomized singular value decomposition (RSVD) has been used to find the dominant subspace for the solution of large scale problems, hence providing an alternative surrogate which in this case inherits the dominant properties of the original problem. For ill-posed problems the surrogate is usually of a sufficient size that its spectral properties still provide a problem that is ill-conditioned. Thus, for both RSVD and Krylov approaches, hybrid methods in which regularization is applied at the subspace level are still required, and depend on the quality and characteristics of the approximated spectral space.

The quality of the spectral space approximation when using the RSVD can be tuned by the use of an oversampling parameter, $p$, that depends on the problem size and the degree of ill-conditioning of the model. Theoretical results quantify the degree by which oversampling is required in order to produce, on the average, sufficiently accurate spectral approximations after truncation of the oversampled space, see e.g. [3]. Our work is motivated by subspace oversampling but as applied for avoiding the semi-convergence issues that arise when using Krylov subspaces of increasing size. Hence, the goal is to find regularized solutions of ill-posed problems which are stable to the choice of the size of the Krylov subspace.

Numerical experiments demonstrate that carefully-determined oversampling of the Krylov space, that is generated using the Golub-Kahan-Bidiagonalization (GKB) as part of the LSQR algorithm, can reduce issues of semi-convergence of the solutions. Specifically, we use $k+p$ steps of GKB to generate the Krylov space, $K_{k+p}$ of size $k+p$, but find the solution using the truncated space of size $k$. It is well-known that with increasing $k$, better estimates of the dominant spectral space of $A$ are achieved; increasing $k$ concentrates the dominant terms of the spectral space with increasing $k$. But, little attention has been paid to investigating the use of spaces of size $k+p$ that are then truncated back to size $k$. In this research, we examine the approximation of the $k+p$ term singular value decomposition, $\tilde{A}_{k+p}$, that is obtained from the GKB algorithm. Given the singular value expansion for $\tilde{A}_{k+p}$, truncating to $k$-terms does not yield $\tilde{A}_k$ that is obtained using just $k$ steps of GKB. The focus of this presentation is thus on the analysis that leads to an effective determination of a reliable oversampling parameter in generating the Krylov space that is to be used in solving a least squares problem. The aim is identify a suitable $p$ directly from the factorization which is developed via the GKB algorithm, namely from $AG_k = H_{k+1}B_k$, where $B_k$ is bidiagonal of size $k+1 \times k$, and $G_k$ and $H_{k+1}$ are column orthogonal matrices with consistent dimensions.

In quantifying the reliability of a rank $k$ approximation of $\tilde{A}_k$ to a matrix $A$, a standard measure employs the quality of the approximation as compared to the best rank $k$ approximation, $\|A_k - A\|_2 = \sigma_{k+1}$, where $A_k$ is the truncated singular value decomposition for $A$ with $k$ terms. A nearest-rank $k$ approximation satisfies $\|\tilde{A}_k - A\|_2 < \frac{1}{2}(\sigma_k + \sigma_{k+1})$. Here $\sigma_i$ are the singular values of
Numerical experiments, all applied with re-orthogonalization of the Krylov basis vectors, demonstrate that the estimate of $\|\tilde{A}_k - A\|_2$ is not sufficient for assessing the overall accuracy of solutions of ill-posed problems. Consideration of the Ritz values and vectors improves the ability to assess the accuracy of approximations, in terms of distances and angles between subspaces, as detailed already in [2], but specifics on how to effectively improve the accuracy of a space of size $k$ from a space of size $k+p$ are not immediate. Moreover, while it is clear that the first $k$ Ritz pairs from a space of size $k+p$ provide better approximations than those from the space of size $k$, when $p > 0$, the quality of this approximation is not immediately seen by examining the convergence to zero of the entries in the bidiagonal matrix that results after $k$ steps of GKB. Thus it is not clear how to find $p$ that is optimal in relation to tuning for computation load and accuracy improvement. In this presentation we will show how the discussion in [5], based on crucial results relating to the spectral gap function, [7], and perturbation bounds presented in [4], can be extended to provide practical means to assess the impact of oversampling, dependent on the conditioning of the problem.

It is also known that solutions of least squares problems obtained with increasing $k$ and $p = 0$ exhibit semi-convergence because the Krylov space inherits the ill-stability of the underlying operator. Semi-convergence also impacts techniques for finding regularization parameters, $\alpha$, automatically. For example, a weighted method of Generalized Cross Validation was presented to tackle the specific issues with GKB solutions of ill-posed problems, [1]. An improved understanding of how to generate a good approximation of a space of size $k$, can then also be applied in the context of determining optimal regularization parameters. Efficiently tuning with increasing subspace size $k$ makes hybrid methods more robust and stabilizes the convergence properties of the solutions.

Numerical examples from image restoration and inversion of three dimensional magnetic data will be presented to support the analysis in the context of solving practical least squares applications. Results contrast approaches using both GKB and RSVD, and demonstrate the impact of involving oversampling with truncation for GKB. To obtain a given accuracy a far smaller space is required when using GKB rather than RSVD, provided that $p$ is appropriately tuned. Moreover, we integrate the analysis into automatically adjusting the regularization parameter $\alpha$ with increasing $k+p$ to efficiently estimate both $k+p$ and $\alpha$.

References


Nonlinearizing two-parameter eigenvalue problems

Emil Ringh, and Elias Jarlebring

Department of Mathematics, KTH Royal Institute of Technology
https://math.kth.se/~eringh/

Abstract

For given matrices $A_1, A_2, A_3 \in \mathbb{C}^{n \times n}$ and $B_1, B_2, B_3 \in \mathbb{C}^{m \times m}$, the two-parameter eigenvalue problem is to find non-trivial quadruplets $(\lambda, x, \mu, y) \in \mathbb{C} \times \mathbb{C}^n \times \mathbb{C} \times \mathbb{C}^m$ such that

\begin{align*}
A_1 x + \lambda A_2 x + \mu A_3 y &= 0 \quad (1a) \\
B_1 y + \lambda B_2 y + \mu B_3 y &= 0. \quad (1b)
\end{align*}

The problem is a special case of the multiparameter eigenvalue problem, which was studied by Atkinson about half a century ago [2]. We assume that the $A$-matrices are large and sparse, and the $B$-matrices are small and dense. We present and analyze a technique to transform the two-parameter eigenvalue problem into an equivalent nonlinear eigenvalue problem (NEP), i.e., a nonlinearization. We characterize the equivalence and use the transformation to derive algorithms for the two-parameter eigenvalue problem. We also connect with research on linearizations for certain NEPs.

Characterization and equivalence of the nonlinearization

The transformation stems from viewing (1b) as a parameterized generalized eigenvalue problem (GEP). More precisely, for a fixed $\lambda \in \mathbb{C}$ solve

\[-(B_1 + \lambda B_2)y = \mu B_3 y \quad (2)\]

for $\mu$ and $y$, and normalize as $c^T y = 1$ for some fixed vector $c \in \mathbb{C}^m$. Due to perturbation theory for eigenvalue problems, there exist families of locally analytic functions $\{g_i\}$ and $\{y_i\}$, that can be evaluated pointwise by solving

\[-(B_1 + \lambda B_2)y_i(\lambda) = g_i(\lambda)B_3 y_i(\lambda) \quad (3a)\]

\[c^T y_i(\lambda) = 1, \quad (3b)\]

for the values of $g_i(\lambda)$ and $y_i(\lambda)$. From the implicit function theorem we get the following result.

**Lemma.** Let $\lambda \in \mathbb{C}$ be given and assume that $(\mu, y)$ is such that (2) is satisfied with $y$ normalized as $c^T y = 1$. Moreover, let $B(\lambda, \mu) := B_1 + \lambda B_2 + \mu B_3$, and assume that the (Jacobian) matrix

\[J(\lambda, \mu, y) := \begin{bmatrix} B(\lambda, \mu) & B_3 y \\ c^T & 0 \end{bmatrix} \quad (4)\]

is nonsingular. Then, there exist functions $g_i : \mathbb{C} \to \mathbb{C}$ and $y_i : \mathbb{C} \to \mathbb{C}^m$ such that

- $g_i$ and $y_i$ are analytic in $\lambda$,
- $g_i$ and $y_i$ satisfy (3) in a neighborhood of $\lambda$,
- $\mu = g_i(\lambda)$ and $y = y_i(\lambda)$.
The nonlinearization of (1) is given by the NEP corresponding to (1a) with \( \mu = g_i(\lambda) \), i.e.,

\[
M(\lambda)x = (A_1 + \lambda A_2 + g_i(\lambda) A_3)x = 0.
\]

(5)

Under the same conditions that the implicit functions exist, we have equivalence between the solutions to the NEP (5) and the two-parameter eigenvalue problem (1).

**Theorem.** Suppose the quadruplet \((\lambda, x, \mu, y) \in \mathbb{C} \times \mathbb{C}^n \times \mathbb{C} \times \mathbb{C}^m\) is such that \( c^T y = 1 \) and \( J(\lambda, \mu, y) \) defined in (4) is nonsingular. Then, \((\lambda, x, \mu, y)\) is a solution to (1) if and only if \((\lambda, x)\) is a solution to the NEP (5) for one pair of functions \((g_i(\lambda), y_i(\lambda)) = (\mu, y)\) which satisfies (3).

Both of the above results relies on the assumption that the Jacobian matrix (4) is nonsingular. However, the Jacobian is singular only in non-generic situations. More precisely, we show that the Jacobian is singular (essentially) if and only if the GEP has a Jordan chain of length two or more.

**Algorithms for the two-parameter eigenvalue problem**

The nonlinearization technique can be used to derive algorithms for the two-parameter eigenvalue problem (1) from algorithms for the NEP (5). Derivatives of \( g_i \) and \( y_i \) can be computed recursively, based on implicit derivation of (3), and projection of the NEP (5) can be interpreted as small-scale two-parameter problems. We also show that if the two-parameter eigenvalue problem is well conditioned [3] and \( g_i \) is computed with a backward stable method, then the error is benign.

**Reversed linearization viewpoint**

It is known that certain NEPs can be linearized, e.g., polynomial and rational. In general, algebraic NEPs can be linearized as pointed out in the presentation [5]. This research connects with the research on linearizations since the two-parameter eigenvalue problem can be linearized using the operator determinants and delta-equations [4, 3]. Hence, the equivalence result can be used to derive linearizations for certain NEPs. Specifically, the NEP \((A_1 + \lambda A_2 \pm \sqrt{(a + \lambda c)(b + \lambda d)} A_3)x = 0\) can be viewed as a nonlinearization of the two-parameter eigenvalue problem (1) for the given matrices

\[
B_1 = \begin{pmatrix} 0 & a \\ b & 0 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 & c \\ d & 0 \end{pmatrix}, \quad B_3 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.
\]

With the notation that \( y = (y_1 \ y_2)^T \), a linearization of the NEP is thus given by

\[
\begin{pmatrix} A_1 & -aA_3 \\ -bA_3 & A_1 \end{pmatrix} \begin{pmatrix} y_1x \\ y_2x \end{pmatrix} = \lambda \begin{pmatrix} -A_2 & cA_3 \\ dA_3 & -A_2 \end{pmatrix} \begin{pmatrix} y_1x \\ y_2x \end{pmatrix}.
\]

For simulation results and illustrations, see the preprint [1]. The software is available online, for both Julia and MATLAB.

**References**


Rational Krylov for Stieltjes matrix functions: convergence and pole selection

Stefano Massei, and Leonardo Robol

Department of Mathematics, University of Pisa
http://people.dm.unipi.it/robol/

Abstract

We are concerned with the evaluation of $x = f(M)v$, where $f(z)$ is a Stieltjes function, which can be expressed in integral form

$$f(z) = \int_0^\infty g(z,t)\mu(t)\,dt, \quad g(z,t) \in \left\{ e^{-zt}, \frac{1}{z+t} \right\}. \quad (1)$$

The two choices for $g(z,t)$ define Laplace-Stieltjes and Cauchy-Stieltjes functions, respectively [3, 7]. The former class is a superset of the latter, and coincides with the set of completely monotonic functions, whose derivatives satisfy $(-1)^j f^{(j)} \geq 0$ over $\mathbb{R}_+$ for any $j \in \mathbb{N}$.

We are interested in two instances of this problem; first, we consider the case $M := A$, where $A \in \mathbb{C}^{n \times n}$ is Hermitian positive definite with spectrum contained in $[a,b]$, $v \in \mathbb{C}^n$ is a generic vector, and a rational Krylov method [6] is used to approximate $x = f(M)v$. In this case, we want to estimate the Euclidean norm of the error $\|x - x_\ell\|_2$, where $x_\ell$ is the approximation returned by $\ell$ steps of the method. Second, we consider

$$M := I \otimes A - B^T \otimes I \in \mathbb{C}^{n^2 \times n^2}, \quad (2)$$

where $A, -B \in \mathbb{C}^{n \times n}$ are Hermitian positive definite with spectra contained in $[a,b]$, $v = \text{vec}(F) \in \mathbb{C}^{n^2}$ is the vectorization of a low-rank matrix $F = U_F V_F^T \in \mathbb{C}^{n \times n}$, and a tensorized rational Krylov method [3] is used for computing $\text{vec}(X) = f(M)\text{vec}(F)$. This problem is a generalization of the solution of a Sylvester equation with a low-rank right hand side, which corresponds to evaluate the function $f(z) = z^{-1}$. Here, we are concerned with estimating the quantity $\|X - X_\ell\|_2$, where $X_\ell$ is the approximation obtained after $\ell$ steps.

We provide a convergence analysis that links the norm of the error for a given choice of poles to a parameter dependent rational approximation (with the given poles) of the kernel functions $e^{-zt}$ and $\frac{1}{z+t}$. These problems can be related — by relying on some Möbius transforms — to the so-called Zolotarev problem on symmetric real intervals [8].

This, in turn, allows us to give explicit expressions of poles that achieve a quasi-optimal convergence rate for the two classes of interest. These results are given for both the standard rational Krylov method and the tensorized version (for $M = I \otimes A - B^T \otimes I$). The analysis for the Cauchy case without Kronecker structure provides analogous results to the ones in [1, Section 6.1] which relies on powerful tools from logarithmic potential theory. See also [5] for similar, although less explicit, results. The results for matrices with Kronecker structure are new, to the best of our knowledge.

It is already known that completely monotonic functions are well approximated by exponential sums [4]. The results that we provide prove constructively that they are also well-approximated by rational functions.

In addition, a byproduct of our analysis is that, in the Kronecker structured case, the low-rank property of the right hand side numerically guarantees the low-rank approximability of the matrix...
This generalizes the well-known property of the solutions of of Sylvester equations with low-rank right hand sides [2]. In particular, we provide bounds for the exponential decay for the singular values of $X$.

To conclude, we discuss a possible application to the solution of fractional diffusion equations on a rectangular domain of the aforementioned rational Krylov method for computing $f(M)v$, when $M$ is Kronecker structured.

References


Numerical Homotopy Methods for Multiparameter Eigenvalue Problems

Lek-Heng Lim, Jose Israel Rodriguez, and Yiling You

Department of Mathematics, University of Wisconsin-Madison

www.math.wisc.edu/~jose/

Abstract

A multiparameter eigenvalue problem (MEP) is defined as follows. Instead of having a single eigenvalue parameter $\lambda$, an MEP has multiple eigenvalue parameters $\Lambda = (\lambda_1, \ldots, \lambda_k)$. We will call

$$H(\Lambda) := A_0 - \lambda_1 A_1 - \cdots - \lambda_k A_k,$$

a linear polynomial matrix in $k$ parameters $\lambda_1, \ldots, \lambda_k$ with matrix coefficients $A_0, \ldots, A_k \in \mathbb{C}^{n \times n}$. For a fixed $k \geq 1$ and given matrices $A_{ij} \in \mathbb{C}^{n_i \times n_j}$ with $j = 0, 1, \ldots, k$, $i = 1, \ldots, k$, consider the linear polynomial matrices

$$H_i(\Lambda) := A_{i0} - \lambda_1 A_{i1} - \lambda_2 A_{i2} - \cdots - \lambda_k A_{ik}, \quad i = 1, \ldots, k.$$

We write $\mathbb{P}^n$ to denote the complex projective $n$-space. The multiparameter eigenvalue problem, or, more precisely, a $k$-parameter eigenvalue problem, is to find $\lambda_1, \ldots, \lambda_k \in \mathbb{C}$ and corresponding $(x_1, \ldots, x_k) \in \mathbb{P}^{n_1-1} \times \cdots \times \mathbb{P}^{n_k-1}$ such that $H_i(\Lambda)x_i = 0, i = 1, \ldots, k$, or equivalently,

$$
\begin{align*}
(\lambda_1 A_{11} + \lambda_2 A_{12} + \ldots + \lambda_k A_{1k})x_1 &= A_{10}x_1, \\
(\lambda_1 A_{21} + \lambda_2 A_{22} + \ldots + \lambda_k A_{2k})x_2 &= A_{20}x_2, \\
& \vdots \\
(\lambda_1 A_{k1} + \lambda_2 A_{k2} + \ldots + \lambda_k A_{kk})x_k &= A_{k0}x_k.
\end{align*}
$$

A solution $(\lambda_1, \ldots, \lambda_k, x_1, \ldots, x_k) \in \mathbb{C}^k \times \mathbb{P}^{n_1-1} \times \cdots \times \mathbb{P}^{n_k-1}$ to the MEP is called an eigenpair, the $k$-tuple $(x_1, \ldots, x_k) \in \mathbb{P}^{n_1-1} \times \cdots \times \mathbb{P}^{n_k-1}$ an eigenvector, and the $k$-tuple $\Lambda = (\lambda_1, \ldots, \lambda_k) \in \mathbb{C}^k$ an eigenvalue. In other words, an eigenvalue $\Lambda \in \mathbb{C}^k$ is a point such that the matrices $H_1(\Lambda), \ldots, H_k(\Lambda)$ are simultaneously singular.

There is a rich mathematical theory behind MEP [1, 2, 8] that places it at the crossroad of linear and multilinear algebra, ordinary and partial differential equations, spectral theory and Sturm–Liouville theory, among other areas. The problem appeared as early as 1836 in the works of Sturm and Liouville on periodic heat flow in a bar, and was studied over the years by many: Klein, Lamé, Heine, Stieltjes, Pell, Carmichael, Bocher, Hilbert among them (see [1, Preface] and [2, Chapter 1]). Furthermore, an MEP encompasses many known types of eigenvalue problems: Standard eigenvalue problems $Ax = \lambda x$; generalized eigenvalue problems (GEPS) $Ax = \lambda Bx$; quadratic eigenvalue problems $(\lambda^2 A + \lambda B + C)x = 0$; polynomial eigenvalue problems $(\lambda^m A_m + \lambda^{m-1} A_{m-1} + \cdots + A_0)x = 0$, may all be reduced to mathematically equivalent MEPs.

There are several approaches that may be used to find an MEP eigenpair. A common approach is to linearize the MEP by constructing a GEP whose eigenvalues correspond to those of the MEP. Its downside is that the size of the GEP can be exceedingly large; in fact finding a linearization that minimizes the size of this GEP is still an active area of research [3, 5, 6]. Our approach is to use a special homotopy continuation method to solve the MEP as it is given in (1), one that exploits the structure of (1).
In this talk, we present the fiber product homotopy method \cite{7} for MEPs, motivated by the notion of fiber products in algebraic geometry. In this method, we do not deform the matrices $A_{ij}$’s as in \cite{4}. Instead, we solve (1) by solving a mathematically equivalent system that we call the fiber product multiparameter eigenvalue problem:

$$H_i(\Lambda_i)x_i = 0, \quad \Lambda_1 = \Lambda_2 = \cdots = \Lambda_k, \quad i = 1, \ldots, k, \quad (2)$$

where $\Lambda_1, \ldots, \Lambda_k$ are to be regarded as different copies of $\Lambda$. While easily seen to be equivalent to (1), the formulation in (2) affords a far more effective homotopy.

For notational simplicity, take $k = 2$; our results in \cite{7} show that the following homotopy $H : \mathbb{C} \times \mathbb{C}^2 \times \mathbb{P}^{n_1-1} \times \mathbb{P}^{n_2-1} \to \mathbb{C}^{n_1+n_2+2}$ can be used to solve (2):

$$H(t, \lambda_1, \lambda_2, \lambda'_1, \lambda'_2, x_1, x_2) = \begin{bmatrix} H_1(\lambda_1, \lambda_2)x_1 \\ H_2(\lambda'_1, \lambda'_2)x_2 \\ (1-t)L_1(\lambda_1, \lambda_2) + t(\lambda_1 - \lambda'_1) \\ (1-t)L_2(\lambda'_1, \lambda'_2) + t(\lambda_2 - \lambda'_2) \end{bmatrix} \quad (3)$$

Here $L_i : \mathbb{C}^2 \to \mathbb{C}$ is a general affine linear function, $i = 1, 2$. More precisely, the start system of our homotopy at $t = 0$, i.e., $H(0, \lambda_1, \lambda_2, \lambda'_1, \lambda'_2, x_1, x_2) = 0$, reduces to two decoupled GEPs

$$H_i(\Lambda_i)x_i = 0, \quad L_i(\Lambda_i) = 0, \quad i = 1, 2,$$

each with $d_i := \deg(\det H_i)$ solutions. Thus the start system has $d_1d_2$ solutions. To track the $d_1d_2$ solutions from $t = 0$ to $t = 1$, we use an Euler–Newton predictor-corrector method where the Euler step gives an approximate eigenpair and the Newton step refines the approximation. At $t = 1$, we obtain the solution to (2) and thus (1). Basic arguments from algebraic geometry allows us to show that this yields all eigenpairs of the multiparameter eigenvalue problem (1) with probability one. The method and results extend to $k > 2$.

To illustrate the effectiveness of the proposed method, we provided extensive numerical experiments in \cite{7} on (i) randomly generated MEPs; (ii) the Mathieu two-parameter eigenvalue problem arising from an elliptic membrane vibration problem; and (iii) singular MEPs, a challenging class of MEPs with a deficiency in the number of eigenpairs that breaks most other methods.

References


Spectral Properties of Doubly-Stochastic Scaling to Uncover Hidden Block Structure

Iain.S. Duff, P.A. Knight, L. le Gorrec, S. Mouysset and D. Ruiz

IRIT, Université de Toulouse

Abstract

1 Introduction

Accurate block detection in matrices is an important tool in many fields of contemporary applied mathematics —examples include block preconditioners for sparse linear systems [1], community detection in networks [2], and data mining [3]—and may involve both sparse and dense matrices. If the underlying block structure is transparent then this may not be a challenging task.

We are developing an algorithm based on algebraic considerations, which is designed to be used in a general setting, in particular where block structure is well hidden. This algorithm is based on interesting spectral properties of the doubly-stochastic scaling of the matrix in which we are looking for numerical block structure. Such a scaling highlights the matrix block structure on a few leading singular vectors, that are then analysed with the help of totally unsupervised tools from signal processing.

2 Doubly stochastic scaling.

We scale a nonnegative matrix $M \in \mathbb{R}^{n \times n}$ into doubly stochastic form by finding diagonal matrices $E, F \in \mathbb{R}^{n \times n}$ so that the row and column sums of $EMF$ are all equal to one.

This scaling has been chosen mostly because of its impact on the spectral elements of $M$. Indeed, a doubly stochastic matrix with a perfect diagonal block structure has leading singular vectors which present a staircase pattern (this a first consequence of the Perron-Frobenius theorem). If the matrices $MM^T$ and $M^TM$ are close to block diagonal in structure then we can expect their leading eigenvectors to present a near staircase pattern once $M$ has been scaled.

We analyse a set of $d \ll n$ leading right and left singular vectors of $M$ to identify its row and column blocks, i.e. the diagonal blocks of $MM^T$ and $M^TM$ respectively.

3 Edges of leading singular vectors.

We highlight the staircase pattern of the singular vectors by sorting their elements in ascending order. It then remains to detect the edges in each vector: these edges indicate the separations between the blocks. We use a well known edge tool from signal processing: a filter—some specially chosen function such as the first derivative of the Gaussian kernel or the Haar function—is applied to the singular vector by means of a convolution product. The peaks in the convolution should coincide with the edges of the vector.
An important filter parameter is its width, which affects the detection of the blocks. Because we presume no a priori knowledge about the blocks, we apply the filter for several widths, and then sum the results.

We apply the edge detection process to every singular vector in the set. If we wish to refine the resultant clustering we can calculate the leading singular vectors in an orthogonal space and then we can attempt to detect new blocks.

4 Quality measure.

After completing the edge detection, we post-process the clustering to deal with two potential issues which may compromise its quality. As a quality measure, we use the numerical version of Newman’s modularity [5] which can be calculated efficiently by exploiting the doubly stochastic nature of $M$.

The first issue is that we may end up with a very fine partitioning of the original matrix which offers few practical benefits. To overcome this we recursively amalgamate pairs of clusters to achieve the best possible improvement of the quality measure.

We also need an effective stopping criterion for the iterative process of SVD followed by block detection. To this end we will only accept a new clustering if it makes a significant improvement to the quality measure, i.e. when the real increase of the quality measure is within a reasonable fraction of the expected increase of an upper-bound.

5 Numerical Experiments

In order to show the versatility of our method, we have applied it on two different clustering tasks: community detection and shape detection in clouds of 2D points. In community detection, the matrix one aims to discover the block structure is sparse, the purpose is then to find dense diagonal blocks by symmetrical permutations of the rows and the columns. In shape detection in clouds of 2D points, we apply our algorithm on the affinity matrix of the dataset, that is a dense matrix, and try to recover numerically dominant diagonal blocks. We compare our method with state-of-the-art on-purpose algorithms, and observe that our totally unsupervised method is competitive (for community detection), if not superior (for shape detection).

References


Iteratively Reweighted FGMRES for Sparse Reconstruction

Silvia Gazzola, James G. Nagy, and Malena Sabaté Landman

Department of Mathematical Sciences, University of Bath (United Kingdom)
https://people.bath.ac.uk/msl39/

Abstract

Large-scale linear ill-posed inverse problems of the form

\[ Ax = b + e, \quad A \in \mathbb{R}^{n \times n} \quad (1) \]

arise in the discretization of problems stemming from various scientific and engineering applications such as astronomical and biomedical imaging. In particular, we are interested in the case where \( A \) is ill-conditioned with ill-determined rank, i.e., the singular values of \( A \) quickly decay and cluster at zero without an evident gap. Due to the noise in the measured data, regularization is needed to obtain a meaningful approximation of \( x \).

A popular scheme to compute approximate solutions of \( x \), when considering Gaussian white noise, is to minimize a least-squares problem with an \( \ell_p \)-norm as a regularization term. This is known to provide a sparse approximation of \( x \) when \( p = 1 \). This problem can be formulated as a non-linear weighted least squares problem of the form:

\[
\min_x \| Ax - b \|_2^2 + \lambda^2 \| W_p(x) x \|_2^2.
\]

A well-established framework to solve this problem, usually referred to as Iteratively Reweighted Least Squares (IRLS) (cf. [5], Chap. 4) or Iteratively Reweighted Norm (IRN) [2], is the local approximation of (2) by a sequence of quadratic problems of the form

\[
\min_{\bar{x}} \| A \bar{x} - b \|_2^2 + \lambda^2 \| W_k(\bar{x}) \|_2^2.
\]

where the sequence of weights \( W_k \) is updated using an available approximate solution from a previous iteration, i.e., \( W_k = W_p(x_k) \). Note that this is a particular instance of a majorization-minimization (MM) scheme [6], where (3) corresponds to the non-constant terms of a particular majorization of the objective function in (2). For large-scale unstructured problems, this method intrinsically relies on an inner-outer iteration scheme that, for example, may exploit the properties of Krylov subspace methods during the inner iterations. It is known that these schemes produce a sequence of approximate solutions that converge to the minimiser of problem (2). However, these methods can be computationally expensive because they require to fully solve a minimization problem at each outer step.

For \( W_k \) square and invertible (this can be assumed through suitable thresholding [6, 2]), we can easily transform problem (3) into standard form, so that

\[
\bar{x}_k = \arg \min_{\bar{x}} \| A W_k^{-1} \bar{x} - b \|_2^2 + \lambda^2 \| \bar{x} \|_2^2 \quad \text{and} \quad x_k = W_k^{-1} \bar{x}_k.
\]

The interpretation of the matrix \( W_k^{-1} \) as a right preconditioner for the least squares problem can be exploited under the framework of prior-conditioning [4]. Indeed, the Flexible Arnoldi decomposition
$AZ_k = V_{k+1} H_k$ can be used to project the least squares problem onto a subspace that adaptively incorporates information about the solution, avoiding the nested cycles of iterations. This idea has been used in FGMRES to find a sparse solution of $\min_x \|Ax - b\|$, both with and without regularization in the projected problem [1]. However, we have proved that the solution of FGMRES does not converge to the solution of (2), and the limit of the sequence of approximated problems cannot be described a priori. Another well justified method to solve problem (2) is the use of the generalized Krylov subspace method for $\ell_p$-$\ell_q$ proposed in [3]. However, this method requires the matrix-vector products with both $A$ and $A^T$ at each iteration.

We propose a new algorithm that, at iteration $k$, updates the weights using the approximate solution from the previous iteration, i.e., $W_{k-1} = W(x_{k-1})$, and benefits from the flexible preconditioning of the new basis vector by $W_{k-1}^{-1}$. Then, it solves the following projected problem to find a new approximation for the solution:

$$\bar{y}_k = \arg \min_{\bar{y}} \|AZ_k \bar{y} - b\|_2^2 + \lambda^2 \|W_{k-1}Z_k \bar{y}\|_2^2 \quad \text{so that} \quad x_k = Z_k \bar{y}_k.$$ (5)

This algorithm, which we refer to as IRW-FGMRES, just requires one matrix vector product per iteration and an additional QR decomposition of a tall and skinny matrix ($W_k Z_k$), which can be performed efficiently. If $\|e\|$ is known, then the regularization parameter can be adaptively set using the discrepancy principle, however, other standard parameter choice strategies can also be used.

Summarizing, this work provides a new algorithm to find a minimizer for problem (2) by partially solving a sequence of quadratic problems of the form (3). It is built upon a solid theoretical justification that guarantees that the sequence of approximate solutions converges to the solution of problem (2), but it has the advantage of building a single flexible Krylov Subspace that encodes regularization through preconditioning. The computational cost for each iteration is dominated by a single matrix-vector product with $A$. The performance of this algorithm will be shown through a variety of numerical examples.

References


SSAI: A symmetric sparse approximate inverse preconditioner for PCG

Shaked Regev and Michael A. Saunders
ICME, Stanford University
stanford.edu/~saunders

Abstract
For Hermitian positive-definite $Ax = b$ with explicit sparse matrix $A \in \mathbb{C}^{n \times n}$, sparse Cholesky factorization may be too expensive. A common alternative is incomplete Cholesky factorization [3, 2]. Matlab’s implementation `ichol` computes a sparse triangular matrix $L$ such that $LL^H \approx A$, for use with the preconditioned conjugate gradient method (PCG). The sparsity of $L + L^H$ is typically a subset of the sparsity of $A$. `ichol` is often effective but sometimes fails.

We consider SSAI [4] as an alternative to incomplete Cholesky for use with a slightly modified form of PCG. SSAI is inspired by the preconditioner developed for GMRES in [5]. Without loss of generality, we apply diagonal scaling to $Ax = b$, so that $A_{jj} = 1$, $j = 1, 2, \ldots, n$. An exact inverse $M$ would make $\|AM - I\|_F^2 = 0$. We estimate each column of $M$ by applying a few iterations of coordinate descent to the problems

$$\min_{m_j} \|Am_j - e_j\|_2^2, \quad j = 1, 2, \ldots, n,$$

where $e_j$ is the $j$th column of the identity. The number of nonzeros in each approximate $m_j$ is limited to the average nonzeros in the columns of $A$, so that $M = [m_1 m_2 \ldots m_n]$ is approximately as sparse as $A$. We set $M \leftarrow (M + M^H)/2$ to obtain an initial preconditioner for PCG.

The resulting $M$ is Hermitian but may not be positive-definite. We therefore monitor the quantity $z_k = Mr_k$ arising in the $k$th iteration of PCG (where $r_k = b - Ax_k$ is the current residual). If $\tilde{\rho} = z_k^H r_k / \|r_k\|^2 < \text{tolM}$, we set $M \leftarrow M + \delta(\text{tolM} - \tilde{\rho})I$ to make $M$ more positive definite, and restart PCG at the current $x_k$. Our experiments use $\text{tolM} = 10^{-2}$, $\delta = 10$ (and require $\|r_k\|/\|b\| \leq \text{tol} = 10^{-8}$). The number of modifications to $M$ is typically 0, 1, or 2.

The columns of $M$ may be computed in parallel, like the columns of `ichol`’s $L$. The matrix-vector products $Mr_k$ may also be parallel, unlike the solves with $L$ and $L^H$. Algorithms for scaling $Ax = b$, computing the SSAI preconditioner $M$, and using $M$ within our modified PCG are given in [4], along with extensive numerical results. On 40 matrices from SuiteSparse [1] with $n$ up to 3 million and $\text{nnz}(A)$ up to 128 million, SSAI is more robust than sparse Cholesky or `ichol`, and is the only method that solved the three largest test problems. On six finite-volume examples from petroleum reservoir simulation with $n \in (400K, 16M)$, `ichol` was more efficient in computing the preconditioner, but PCG + SSAI was significantly more efficient in solving $Ax = b$. On four finite-element examples modeling Antarctic ice flow with $n \in (600K, 5M)$, SSAI succeeded with 1, 2, 3, 2 modifications to $M$, but `ichol` failed on all cases.

References


Computing the closest real normal matrix and the normal completion

Nicola Guglielmi and Carmela Scalone

Dipartimento di Ingegneria e Scienze dell’Informazione e Matematica, University of L’Aquila

Abstract

We consider the problems of computing the nearest normal matrix $X$ to a given non-normal matrix $A \in \mathbb{C}^{n \times n}$, under certain constraints, namely:

(i) if $A$ is real, we also impose that $X$ is real; i.e. we aim to find a matrix $X$ which solves

$$||X - A||_F \rightarrow \min, \quad X \in \mathcal{N}_{n \times n}^R.$$  \hspace{1cm} (1)

where $\mathcal{N}_{n \times n}^R$ denotes the set of real normal matrices.

(ii) if $A$ has known entries on a given sparsity pattern $\Omega$ and unknown/uncertain entries otherwise, we aim to find a normal completion $X$ of $A$, which is equivalent to solve

$$||P_{\Omega}(X - A)||_F \rightarrow \min, \quad X \in \mathcal{N}^{n \times n} \text{ or } \mathcal{N}_{n \times n}^R.$$ \hspace{1cm} (2)

where $\mathcal{N}^{n \times n}$ is the set of normal matrices and $P_{\Omega}$ is the orthogonal projection (with respect to the Frobenius metrics) onto the the complement of the sparsity pattern $\Omega$.

When all the entries of $A$ are modifiable, there exists an algorithm by Ruhe [1], which is able to compute the closest normal matrix. However, nothing in Ruhe’s algorithm guarantees the reality of the solution if $A$ is real and a closed form solution exists just for the case $n = 2$ (see [2]). For such reasons, we are motivated to develop a numerical strategy able to preserve reality for finding the nearest normal matrix. Linear algebra literature is very rich of characterizations of normality. In particular, we consider the well-defined scalar measure of normality by Henrici [3]

$$\nu(A) = ||A||_F^2 - \sum_{i=1}^{n} |\lambda_i(A)|^2$$

This characterization allows us to formulate as equivalent problem the minimization of a functional of an unknown matrix, which should be normal, fulfil the required constraints and have minimal distance from the given matrix $A$. The method to solve these particular matrix nearness problems works on two levels.

Given a square matrix $A \in \mathbb{C}^{n \times n}$ (or $\mathbb{R}^{n \times n}$) and a pattern $\Omega$, for a fixed value $\varepsilon > 0$, we start considering the functional, measuring the distance to normality of the perturbed matrix $A + \varepsilon E$,

$$\nu_{\varepsilon}(E) = \frac{1}{2}||A + \varepsilon E||_F^2 - \frac{1}{2} \sum_{i=1}^{n} |\lambda_i(A + \varepsilon E)|^2$$ \hspace{1cm} (3)

where $E = (e_{ij}) \in \mathbb{C}^{n,n}$ ($E \in \mathbb{R}^{n,n}$ if $A$ is real) is a matrix of unit norm $||E||_F = 1$ such that $P_{\Omega}(E) = E$, that is

$e_{ij} = 0$ if $(i,j) \in \Omega$. 

354
In the Inner Level of the method, we fix $\varepsilon > 0$ and minimize $\nu_\varepsilon(E)$ over all matrices $E$ of unit Frobenius norm, yielding a minimizer $E(\varepsilon)$.

For $\nu(\varepsilon) = \nu_\varepsilon(E(\varepsilon))$, the problem of the Outer Level consists in looking for the smallest value $\varepsilon^*$ such that $\nu(\varepsilon^*)$ is minimized. To solve the problem at the inner level we compute the gradient of the functional (3) and employ a descent method. Then, for every $\varepsilon > 0$, we obtain a perturbation $E(\varepsilon)$, for which the distance to normality of the associated perturbed matrix $A + \varepsilon E(\varepsilon)$ is as small as possible. Finally, considering the scalar function

$$\varepsilon \rightarrow \nu(\varepsilon) = \nu_\varepsilon(E(\varepsilon))$$

we have solve the problem

$$\min\{\varepsilon > 0 : \nu(\varepsilon) \to \min\},$$

which identifies $\min_\varepsilon \nu(\varepsilon) = 0$ as the distance to normality of $A$.

References


Recent progress on nonlinear eigenvalue problems with algebraic branch points

Zhaojun Bai, Weiguo Gao, Xin Huang, Meiyue Shao, Roel Van Beeumen, and Chao Yang

School of Data Science, Fudan University
https://people.cs.umu.se/~myshao/

Abstract

The gun problem [2, 3] is a well-known nonlinear eigenvalue problem of the form

\[
\left( K - \lambda M + i \sqrt{\lambda - \sigma_1^2} W_1 + i \sqrt{\lambda - \sigma_2^2} W_2 \right) x = 0, \quad x \in \mathbb{C}^N \setminus \{0\},
\]

which attracts a lot of attention since the problem was proposed. Most existing methods for solving (1) are approximation-based—first approximate the square roots by rational functions and then linearized the resulting rational eigenvalue problem. These methods have intrinsic difficulties finding eigenvalues at or close to branch points, because a square root function is not analytic in any neighborhood of its branch point. Even for a branch point free region, when the branch point is not far away from the boundary of the region, it typically requires a high degree in order for a rational function to approximate a square root function accurately. This often leads to high computational complexity, slow convergence, as well as numerical instability in the subsequent eigensolver. Therefore, specialized techniques are required to handle square roots, or more general functions with branch points. We will focus on two algebraic techniques.

One technique [1] is to directly linearize nonlinear eigenvalue problems with square roots. For example, the gun problem (1) can be transformed to the following linear eigenvalue problem:

\[
\begin{bmatrix}
K - \lambda M & (\sigma_1^2 - \lambda) W_1 & (\sigma_2^2 - \lambda) W_2 & 0 \\
W_1 & K - \lambda M & 0 & (\sigma_2^2 - \lambda) W_2 \\
W_2 & 0 & K - \lambda M & (\sigma_1^2 - \lambda) W_1 \\
0 & W_2 & W_1 & K - \lambda M
\end{bmatrix}
\begin{bmatrix}
- \sqrt{\lambda - \sigma_1^2} \sqrt{\lambda - \sigma_2^2} x \\
\i \sqrt{\lambda - \sigma_2^2} x \\
\i \sqrt{\lambda - \sigma_1^2} x \\
x
\end{bmatrix} = 0.
\]

Such a linearization is possible because square roots are algebraic functions (i.e., global analytic functions which are algebraic elements over the field \( \mathbb{C}[\lambda] \)). A general algebraic nonlinear eigenvalue problem (ANEP) is an eigenvalue problem of the form

\[
T(\lambda)x = 0, \quad x \in \mathbb{C}^n \setminus \{0\},
\]

where \( T(\lambda) \) is an \( N \times N \) \( \lambda \)-matrix whose entries are algebraic functions of \( \lambda \). Since the set of algebraic functions forms a field, in theory algebraic nonlinear eigenvalue problems can always be linearized. Due to the algebraic closedness of this field, ANEP is also the largest class of nonlinear eigenvalue problems that can be linearized by purely algebraic transformations.

In practice, linearization of ANEP can be achieved through companion forms. For instance, for \( f(\lambda) \) being algebraic with minimal polynomial \( (f(\lambda))^n + \sum_{k=0}^{n-1} a_k(\lambda) (f(\lambda))^k \), the ANEP

\[
(A(\lambda) + f(\lambda)B)x = 0,
\]

is
which involves \( f(\lambda) \), can be linearized as

\[
\begin{bmatrix}
A(\lambda) - a_{n-1}(\lambda)B & -a_{n-2}(\lambda)B & \cdots & -a_1(\lambda)B & -a_0(\lambda)B \\
B & A(\lambda) & & & \\
& \ddots & \ddots & & \\
B & B & A(\lambda) & & \\
B & & & A(\lambda) & \\
\end{bmatrix}
\begin{bmatrix}
x f^{n-1}(\lambda) \\
x f^{n-2}(\lambda) \\
\vdots \\
x f(\lambda) \\
x \\
\end{bmatrix} = 0,
\]

with \( f(\lambda) \) being eliminated from the \( \lambda \)-matrix. ANEPs with multiple “irrational” functions can be linearized by repetitively applying this elimination process. This type of linearizations is robust and will not miss any eigenvalue at or close to branch points. In addition, it can be shown that algebraic multiplicity of each eigenvalue is preserved in a certain sense. For ANEPs involving only a few “irrational” with low degree, this type of linearizations is useful for both theoretical analysis and practical computation [4].

There is also another technique [5] tailored to handle square roots. It aims at finding eigenvalues in a certain region of interest. For a square root function \( f(\lambda) = \sqrt{\lambda - \lambda_0} \), a simple change of variable \( \lambda = \lambda_0 + \mu^2 \) eliminates the branch point and obtains an analytic function in \( \mu \). In fact, this type of transformations is applicable to more general functions with algebraic branch points: for a global analytic function \( f(\lambda) \) with an algebraic branch point of degree \( p - 1 \) at \( \lambda_0 \), a change of variable \( \lambda = \lambda_0 + \mu^p \) suffices to eliminate the branch point since in a neighborhood of \( \lambda_0 \) the function \( f(\lambda) \) can be expanded as a Laurent series in terms of \( \mu = \sqrt[2p]{\lambda - \lambda_0} \). After this transformation, the function becomes analytic in \( \mu \) and then can be more easily approximated by a rational function. When a nonlinear eigenvalue problem involves multiple functions with algebraic branch points, we can always divide the region of interest into a few subregions, each with only a single branch point inside it. By carefully choosing branch cuts from other branch points, each subregion can be made branch cut free, and then the problem in each subregion can be solved by applying an appropriate transformation. Numerical experiments demonstrates that after eliminating branch points with appropriate transformations, the resulting analytic functions can be well approximated by low degree rational functions, leading to low computational complexity and faster convergence in the subsequent eigensolver.

References


On computing the STRUCTURED DISTANCE TO SINGULARITY for matrix pencils using optimization

Punit Sharma

Department of Mathematics, Indian Institute of Technology Delhi, India

http://web.iitd.ac.in/~punit1718/

Abstract

Let \( L(z) = A + zB \) be a matrix pencil where \( A \) and \( B \) are \( n \times n \) complex matrices, then \( L(z) \) is regular if \( A + \lambda B \) is invertible for some \( \lambda \in \mathbb{C} \). A matrix pencil which is not regular is often called as singular matrix pencil. The matrix pencil \( L(z) \) is said to be structured if \( (A,B) \) belong to a special subset \( S \) of \((\mathbb{C}^{n,n})^2\). We consider Hermitian, skew-Hermitian, *-even, *-odd, *-palindromic and *-antipalindromic structures.

Given a regular structured matrix pencil \( A + zB \), where \( (A,B) \in S \), the problem of determining the nearest singular pencil \( \Delta_A + z\Delta_B \), where \( (\Delta_A, \Delta_B) \in S \) with respect to some specific norm is called the structured distance to singularity for matrix pencil \( A + zB \). If \( S = (\mathbb{C}^{n,n})^2 \) then it is called unstructured distance to singularity for pencil \( A + zB \). Due to its great importance in many engineering applications, this problem has attracted many mathematicians in last two decades and is still an open problem [3]. As an example, in the linear time-invariant control systems

\[
E \dot{x} = Ax + By, \quad y = Cx,
\]

if the matrix pencil \( A + zE \) is singular, then the initial value problem of solving (1) with a consistent initial value \( x_0 \) is not solvable or the solution is not unique. Thus the system equation (1) is not well posed if \( A + zE \) is singular. An unfortunate choice of linear feedback control may make the system singular or nearly singular. Further, eigenvalues and eigenvectors of a regular matrix pencil may vary discontinuously in the neighborhood of a singular pencil. Thus, in such cases sensitivity of the problem is closely related to the nearest ill posed problem [4], and some special numerical methods are required [5]. If the matrix pencil with additional structures is considered, then the use of structure preserving algorithms is advisable.

Many heuristic bounds are available in the literature [3] for the distance to singularity. Recently a differential equation based approach has been developed in [6], that tightly bounds the distance to singularity from above. However, there is no better algebraic approach known to compute this distance.

An equivalent characterization for singular pencils is that a pencil \( A + zB \), \( A, B \in \mathbb{C}^{n,n} \) is singular if there exist \( (\lambda_1, \ldots, \lambda_{n+1}) \in \mathbb{C}^{n+1} \) such that \( \lambda_i \neq \lambda_j \) if \( i \neq j \), and \( \det(A + \lambda_i B) = 0 \) for \( i = 1, \ldots, n+1 \). Thus the distance to singularity can be reformulated in terms of backward error of making any distinct \( n + 1 \) points in the complex plane as approximate eigenvalues of \( A + zB \) at once. Explicit formulas for structured eigenvalue backward errors have been obtained for matrix pencils with Hermitian and related structures in [1], and for palindromic structures in [2].

Motivated by the above considerations, we briefly present some of the main results in [1] and [2]. We then present a purely linear algebra based approach to obtain computable formulas to determine tight bounds for the unstructured and structured distances to singularity. These formulas involve minimizing the largest eigenvalue of a parameter depending Hermitian matrix which can be computed by using a suitable optimization method. The formulas have been obtained with respect to the norm \( \sqrt{\|A\|^2_2 + \|B\|^2_2} \), where \( \| \cdot \|_2 \) is the matrix spectral norm.
A pencil can also be singular due to a common left or right null vector. We develop explicit computable formula to compute the distance to the nearest structured pencil \((A + \Delta_A) + z(B + \Delta_B)\) such that \((A + \Delta_A)\) and \((B + \Delta_B)\) have a common left or right null vector. This gives an upperbound for the distance to singularity as in general such a situation need not occur for singular pencils.

We also present some numerical experiments to compare the distance to singularity obtained in this work with the one in [3] and [6]. Finally, we note that this approach extends to structured matrix polynomials of higher degrees.

References


Improved Algorithms and Parallel Software for CP Decomposition

Linjian Ma, Navjot Singh, and Edgar Solomonik

Department of Computer Science, University of Illinois at Urbana-Champaign
http://solomonik.cs.illinois.edu

Abstract

Tensor decompositions have numerous applications within scientific computing and data analysis. The canonical polyadic (CP) decomposition [1] is the perhaps the most prominent of these. This decomposition can be used for discovery of fast bilinear algorithms [2], obtaining reduced scaling quantum chemistry algorithms [3], as well as compressing and understanding datasets from various domains [4]. We describe a new algorithm for CP decomposition, pairwise perturbation, which improves state of the art approaches in theory and practice. Further, we provide a comparative study of the accuracy and performance (in sequential and distributed settings) of various CP decomposition methods. We identify a new robust regularization approach and provide the first effective parallelizations of some of these methods for dense and sparse tensors. We also describe a new approach for tensor completion with CP decomposition, which yields the first high-level distributed algorithm implementations for this problem. Finally, we provide a software library with high-level Python implementations of many tensor decomposition algorithms, which can run sequentially or leverage the Cyclops library for parallel execution.

CP decomposition seeks to approximate a tensor via a sum of tensor products, which are columns of factor matrices. For example, given an $s \times s \times s$ tensor $T$, its rank-$R$ CP decomposition is given by $t_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr}$. The most common algorithm to compute a CP decomposition is alternating least squares (ALS), which updates $A$, $B$, and $C$ in an alternating manner by solving quadratic optimization problems. A sweep of ALS computes the set of $N$ quadratic optimization problems for an order $N$ tensor via the normal equations, using dimension trees to amortize intermediates with total cost $O(s^2NR + NR^3)$. We propose the pairwise perturbation algorithm [5], which performs a approximate updates to the right-hand-sides of each set of normal equations, achieving a total cost of $O(s^2R + NR^3)$ per approximate ALS sweep. We provide theoretical bounds on the approximation error of pairwise perturbation, show that this approximate sweep can be orders of magnitude faster than an exact sweep, and demonstrate that the technique accelerates ALS in ‘swamp’ regimes of the optimization space, typically yielding 2-3X improvements in overall execution time.

The Gauss-Newton method for nonlinear least squares is a competitive alternative to ALS for CP decomposition [6]. This method also solves a linear least squares problem at each iteration, but this quadratic approximation now updates all factor matrices ($A$, $B$, and $C$) at once. ALS generally achieves linear monotonic convergence, while Gauss-Newton lacks monotonicity but can achieve quadratic convergence. We provide new convergence studies that demonstrate that Gauss-Newton is much more effective at finding exact CP decompositions as well as at computing better approximations than ALS [7]. Further, we propose a regularization strategy that leads to faster and more probable convergence to desirable minima. While accurate, naive use of the Gauss-Newton method for CP decomposition requires inverting or factorizing the approximate Hessian, which has a memory footprint of $O(N^2s^2R^2)$ and a cost of $O(N^3s^3R^3)$. However, the use of the implicit structure of the Hessian makes possible computation of matrix-vector products with cost $O(N^2sR^2)$, enabling efficient optimization via the conjugate-gradient (CG) method [8]. We provide the first parallelization of this approach and demonstrate that despite having less parallelism than ALS, Gauss-Newton method can outperform ALS in large-scale calculations [7].
The ALS and Gauss-Newton methods are applicable for decomposition of both dense and sparse tensors. However, the computational challenges change substantially when CP decomposition is used for tensor completion, a generalization of the matrix completion problem, which is widely studied in machine learning. For tensor completion, the goal is to find a reduced model for a partially observed dataset, in this context a set of entries $t_{ijk}$. The most common approaches for tensor completion are ALS, coordinate descent, and stochastic gradient descent. ALS has been shown to generally converge most rapidly in both iterations and overall time in a variety of previous studies. However, in ALS for tensor completion, unlike in sparse tensor decomposition, a different set of linear least squares equations (not just right-hand-sides) is needed for each row of the factor matrices. This complication makes the computation of these equations and parallelization thereof substantially more difficult. We propose the use of batched implicit CG for this problem, which makes possible the expression of the equations in the ALS tensor completion algorithm in terms of the tensor contractions,

$$y_r^{(i)} = \sum_{j=1}^{s} \sum_{k=1}^{s} \sum_{t=1}^{R} b_{jr} c_{kr} \Omega_{ijk} b_{jt} c_{kt} x_t^{(i)},$$

where $\Omega_{ijk}$ is a sparse tensor of binary values indicating presence of observed values, while $x_t^{(i)}$ and $y_r^{(i)}$ are the input and result to matrix-vectors product within CG, which update the $i$th row of the matrix $A$ within a quadratic optimization subproblem in ALS. We introduce a new multi-tensor contraction primitive that computes a pointwise product between a tensor and tensor products (TTTP), which can be computed much more efficiently than pairwise tensor contractions. In combination with pairwise sparse tensor contractions or an efficient MTTKRP kernel, TTTP makes possible efficient execution of the above implicit matrix-vector product expression [9]. This primitive also accelerates residual computation in tensor completion and sparse tensor decomposition.

We provide parallel performance studies of the above tensor decomposition and tensor completion algorithms on the Stampede2 supercomputer by leveraging the Cyclops library for tensor computations [10]. All software is available via https://github.com/cyclops-community.

References


A unifying framework for recycling-based iterative methods

Kirk M. Soodhalter, Eric de Sturler, and Misha Kilmer

Trinity College Dublin, the University of Dublin
https://math.soodhalter.com

Abstract

In many applications in the computational sciences, there is a need to solve not one but many hundreds or thousands of large-scale linear systems

\[ A^{(i)} x^{(i)} = b^{(i)} \quad i = 1, 2, \ldots \]

to achieve a desired goal. Consecutive coefficient matrices are generally considered to be "close" in some way. Subspace recycling methods are a popular class of iterative methods enabling effective reuse of subspace information generated during the run of an iterative method (usually a Krylov subspace method) applied to \( A^{(i+1)} x^{(i+1)} = b^{(i+1)} \). There have been many manifestations of methods which seek to augment an existing Krylov subspace with some additional vectors in order to accelerate convergence.

The first such method to bear the "recycling" moniker was based on the nested inner-outer GCRO-type methods [7] where this technique was combined with the deflated restart (DR) strategy first proposed in [5] leading to the proposal of the GCRO-DR method in [6], which is a flavor of GMRES with subspace recycling. Recycled versions of MINRES [9] and various bi-orthogonal Lanczos-based methods (e.g., [1]) have also been proposed, and a version of conjugate gradients which fits into this category was proposed a few years prior [8], but without the recycling aspect. In the ill-posed problems settings, similar methods were also proposed independently.

This work concerns the proposal of a general framework encompassing most such methods for both the well- and ill-posed problems settings. This framework will not only be useful for analysis purposes; we aim to leverage the knowledge gained through it to enable the systematic design and proposal of custom recycling methods which are tailored to, e.g., specific problems or an underlying Krylov subspace method. Previous authors have proposed a framework for both optimal methods, such as minimum residual and CG methods,[2, 3] and for non-optimal methods based on the biorthogonal Lanczos method [4]. We build upon and extend these ideas to get something more general and flexible to suit our goals.

Most Krylov subspace methods can be derived according to a correction/constraint formulation. For general subspaces \( W \) and \( \tilde{W} \) of equal dimension, this can be expressed as

select \( t \in W \) such that \( r_0 - At \perp \tilde{W} \),

which can also be described by the residual projection \( b - Ax_j = r_0 - At = (I - Q)r_0 \), where \( Q \) is a projector onto \( AW \), where \( r_0 = b - A x_0 \). Should this projection of the residual be equivalent to the minimization of an error functional, then the projection is orthogonal with respect to some inner product; otherwise it is oblique.

We approach the development of our framework for recycled methods from this direction. We have a subspace \( U \) of dimension \( k \) which is the recycled subspace. Additionally, we iteratively generate a (Krylov) subspace \( V_j \) of dimension \( j \) at iteration \( j \). We also require sibling spaces with the same
dimensions of $k$ and $j$, respectively, $\tilde{U}$ and $\tilde{V}_j$. This leads to the general Petrov-Galerkin formulation for a recycled method

$$\text{select } s \in U \text{ and } t \in V_j \text{ such that } r_0 - A(s + t) \perp \tilde{U} + \tilde{V}_j.$$ (1)

There exist projectors $P$ ($Pw \in U$) and $Q$ ($Qw \in A U$) with $QA = AP$ such that enforcing (1) is equivalent to making a Petrov-Galerkin approximation for the projected problem

$$(I - Q)A t = (I - Q)r_0$$ (2)

with correction space $V_j$ and constraint space $\tilde{V}_j$ (i.e., we apply some iterative method directly to the projected problem). This produces approximation $t_j \in V_j$, and the full approximation satisfying (1) is $x_j = x_0 + Pe_0 + (I - P)t_j$, where $Pe_0$ is a computable projection of the initial error. Furthermore, the residual $b - Ax_j$ agrees with the residual produced by applying the iterative method directly to (2), meaning the projected operator and right-hand side tell us a great deal about the behavior of the method.

Versions of this result have been proven for a number of cases. The strength of our more general formulation, however, is that it separates the the choices of $V_j$ and $\tilde{V}_j$ from the projected operator $(I - Q)A$ and projected right-hand side $(I - Q)r_0$. This generalization allows us to choose other projected or non-projected Krylov subspaces to use for $V_j$, meaning most recycled methods in the literature can be described by this single framework. That has not been true previously. This has implications for our ability to analyze existing methods. More importantly, though, is that this gives us the flexibility to produce new customized recycled methods based on algorithmic or application needs, and we can approach the design of new methods in a standardized way. The framework is also fully compatible with rectangular systems and thus Golub-Kahan-based iterative schemes.

In this talk, we will present the details of the new framework. We will demonstrate how one can use the framework to systematically create new recycled methods based on specific constraints or built on top of existing Krylov subspace methods. We will further show that the framework can be used to design augmented methods built on Krylov subspace methods which do not use a residual projection constraint.

References


Spectral information in second order differential operators and cost of iterative computations

E. Carson, T. Gergelits, K.-A. Mardal, B. F. Nielsen, and Z. Strakoš

Nečas Center for Mathematical Modeling, Charles University, Prague
www.karlin.mff.cuni.cz/~strakos/

Abstract

The cost of iterative algebraic computations in solving problems described by partial differential equations (PDEs) is typically linked with some simplified spectral information about the discretized preconditioned operators. This contribution will revisit and challenge some common practices using a practically important class of PDEs and the conjugate gradient method.

In the paper [1] Nielsen, Hackbusch and Tveito study the operator generated by using the inverse of the Laplacian as preconditioner for second order elliptic PDEs $-\nabla \cdot (k(x,y)\nabla u) = f$ with the uniformly positive function $k(x,y)$. They prove that the range of $k(x,y)$ is contained in the spectrum of the preconditioned operator, provided that the scalar function $k(x,y)$ is continuous. Their rigorous analysis only addresses mappings defined on infinite dimensional spaces, but the numerical experiments in the paper suggest that a similar property holds in the discrete case.

Motivated by this investigation, the recent paper [2] analyzes the eigenvalues of the matrix $L^{-1}A$, where $L$ and $A$ are the stiffness matrices associated with the Laplace operator and second order elliptic operator with a uniformly positive scalar coefficient function, respectively. Using only technical assumptions on $k(x,y)$, it is proved that there exists a one-to-one pairing between the eigenvalues of $L^{-1}A$ and the intervals determined by the images under $k(x,y)$ of the supports of the finite element nodal basis functions. As a consequence, one can show that the nodal values of $k(x,y)$ yield accurate approximations of the eigenvalues of $L^{-1}A$. This is, to my knowledge, the first published theoretical result on the topic. The proof of the main theorem combines in an elegant way an operator perturbation argument with the classical combinatorial Hall’s theorem for bipartite graphs.

As described in [3], the results from [1] and [2] can be further generalized and completed in several ways. First, for functions $k(x,y)$ continuous over the whole domain (not necessarily uniformly positive) the spectrum of the preconditioned infinite dimensional operator is equal to the range of the function $k(x,y)$. Second, if the continuous function $k(x,y)$ is replaced by a two-dimensional continuous tensor $K(x,y)$ with the spectral decomposition $K = Q\Lambda Q^*$, then the spectrum of the infinite dimensional operator is equal to the minimal interval containing the ranges of the function entries of the diagonal matrix $\Lambda(x,y)$. Note that this generalization is nontrivial because the individual ranges of the diagonal function entries of $\Lambda$ can be disjoint. Third, generalization to three dimensions depends on the existence of the continuous spectral decomposition of $K(x,y)$. The rest of the proof of the consequences for the spectrum of the preconditioned discretized problems will follow the path of [2].

The presented theoretical results have immediate relevance for understanding how the convergence of the conjugate gradient method may depend on the whole spectrum of the preconditioned matrix. We will provide evidence explaining the role of the large outlying eigenvalues. We will explain in which cases and why the common argument about clustering of eigenvalues can be relevant for description of convergence and, even more importantly, when and why its use leads to wrong conclusions.
The contribution will end with discussion of the related recent attempts (see, e.g., [4]) to build up a general ‘complexity theory’ that aims at including iterative computations; see [5].

References


Simultaneous Diagonalization of Nearly Commuting Hermitian Matrices:
Do-one-then-do-the-other

Brian D. Sutton

Department of Mathematics, Randolph–Macon College
https://folios.rmc.edu/bsutton/

Abstract

Commuting Hermitian matrices $A$ and $B$ may be simultaneously diagonalized by a common eigenvector matrix $Q$:

$$Q^H A Q = \text{diag}(\lambda_1, \ldots, \lambda_n), \quad Q^H B Q = \text{diag}(\mu_1, \ldots, \mu_n).$$

In finite precision, though, $A$ and $B$ likely do not commute exactly, and therefore a numerical algorithm must bring the eigenvectors of $A$ and $B$ into alignment.

We present a new algorithm for simultaneous diagonalization, which is supported by new perturbation results for nearly commuting Hermitian matrices. We call the new algorithm Do-one-then-do-the-other. It is backward stable and efficient, and because it relies on a black-box eigenvalue solver, it can be tailored to a wide range of computer architectures.

An existing algorithm for simultaneous diagonalization is the Jacobi-like algorithm of Bunse-Gerstner, Byers, and Mehrmann [1]. It has received significant attention, particularly for an application to beamforming.

The 1993 article introducing the Jacobi-like method illustrates the difficulty of simultaneous diagonalization through an algorithm called diagonalize-one-then-diagonalize-the-other (the original “DODO” method). Suppose $A$ and $B$ commute exactly and that arithmetic is exact, and let $Q$ be an eigenvector matrix for $A$. Then $Q^H B Q$ is diagonal unless $A$ has any repeated eigenvalues. In that case, an eigenvalue decomposition for $Q^H B Q$ can resolve eigenvectors within the eigenspaces of $A$ and complete the simultaneous diagonalization. Unsurprisingly, this scheme fails in finite precision when eigenvalue clusters are present [1].

I began my recent work with the goal of developing a spectral divide-and-conquer method for simultaneous diagonalization. Eventually, though, I realized that I had discovered a numerically stable variant of the alternating method described above.

Let me start by describing a new spectral divide-and-conquer method. First, a large eigenvalue gap is located—large relative to the spectrum spreads $\lambda_{\text{max}} - \lambda_{\text{min}}$ and $\mu_{\text{max}} - \mu_{\text{min}}$. The relevant spectrum is split at the gap, invariant subspaces are computed, and $A$ and $B$ are simultaneously reduced to block-diagonal form:

$$
\begin{bmatrix}
A_{11} & 0 \\
0 & A_{22}
\end{bmatrix}
\quad \text{and} \quad
\begin{bmatrix}
B_{11} & 0 \\
0 & B_{22}
\end{bmatrix}.
$$

Then this process is repeated recursively on the diagonal blocks. We prove that the reduction is backward stable even though invariant subspaces can be ill conditioned. As the recursion proceeds, the algorithm is forced to split at smaller and smaller gaps, but at the same time less and less accuracy is required because the blocks look more and more like multiples of identity matrices. Our analysis is based on an apparently novel measure incorporating an eigenvalue gap and a spectrum spread, and the results transition smoothly from the well-conditioned case to the ill-conditioned case.

The spectral divide-and-conquer method seeks large eigenvalue gaps. But how can a large gap be found reliably and inexpensively? Our solution is to amortize the cost of gap calculations by
computing all \( n \) eigenvalues of \( A \) or \( B \) at once. Then we split the spectrum not just at one gap but simultaneously at every large gap. If any nontrivial blocks remain after reduction to block-diagonal form, then the process is repeated recursively, leading to a back-and-forth process in which eigenvectors from \( A \) are applied to \( B \) and vice versa. We find ourselves back at (essentially) the diagonalize-one-then-diagonalize-the-other scheme described in the paper by Bunse-Gerstner et al., but with a rule for classifying eigenvalue clusters and the necessary perturbation theory to guarantee numerical stability.

The computation and communication costs of the new method are dominated by an \( n \)-by-\( n \) eigenvalue decomposition. Although the analysis is inspired by spectral divide-and-conquer, the implementation relies on a black-box eigenvalue solver, which can be selected to fit any desired computer architecture.

One final note: A subtle point in the analysis concerns the recursive step, when tiny entries in off-diagonal blocks are replaced by zeros. In other divide-and-conquer algorithms, this action is a triviality, but in this problem, truncation threatens to destroy commutativity. Is it true that nearly commuting, nearly block-diagonal matrices have diagonal blocks that nearly commute? This question leads us into a substantial literature on the difference between “almost commuting” and “nearly commuting,” most notably a result of Pearcy and Shields [2]. Eventually we find, yes, the tiny entries may be safely dropped.

In conclusion, we have a fast, flexible, backward-stable algorithm for simultaneous diagonalization, supported by new results in perturbation theory.

References


Asynchronous Schwarz Methods: Theory and Computations

Daniel B. Szyld

Department of Mathematics, Temple University, Philadelphia
math.temple.edu/szyld

Abstract

In this talk I will present an overview of the latest developments on asynchronous Schwarz methods for the numerical solution of linear systems of equations, and in particular those that arise from the discretizations of PDEs. These methods have been shown to be particularly effective for very large problems on massively parallel architectures, and they have the potential to continue to be effective in the forthcoming exascale environments.

Asynchronous iterations were extensively studied in the last two decades of the 20th Century [6]. The convergence theory is based on iterations on product spaces, essentially Block Jacobi-type methods. Using Restricted Additive Schwarz, in its optimized version, and with overlap between the subdomains (or blocks of variables), gives a very fast algorithm [1, 3, 4], and the theory of asynchronous iterations can apply. In fact, convergence results for several situations have been obtained, including for the shifted Laplacian on the whole plane [8], and on bounded domains with different configurations of the subdomains [2, 7]. Different tools from functional analysis are used in the proofs of these results.

Asynchronous Schwarz methods are fast because each processor works on the variables corresponding to a block (or subdomain) and after the solution of the “local” problem, it can start the new update without having to wait for all processors to finish, and for all the information to go from each processor to the next. In this manner, global communication is completely avoided. Communication takes place while the computations continue. The solution of the local problem is performed with whatever information has been received from the other processors at the start of this local computation. In addition, unlike Krylov subspace methods, no inner products are needed, so that no global communication is necessary.

One can further accelerate these computations by the judicious addition of an additive course grid correction. (Multiplicative corrections are out of the question, since we do not want sequential work.) In this manner a scalable algorithm is obtained [5]. Theory for this asynchronous two-level method was also developed. Experiments will be reported with about 160 million unknowns, corresponding to a 3D problem, showing that the synchronous version of the method convergences in about 1/5 of the time of the synchronous counterpart.

References


The Conditioning of Least Squares Problems in Variational Data Assimilation

Jemima M. Tabeart\textsuperscript{1}, Sarah L. Dance\textsuperscript{2}, Amos S. Lawless\textsuperscript{2}, Nancy K. Nichols\textsuperscript{2} and Joanne A. Waller\textsuperscript{3}

\textsuperscript{1}School of Mathematics, University of Edinburgh, UK
\textsuperscript{2}School of Mathematical, Physical and Computational Sciences, University of Reading, UK
\textsuperscript{3}MetOffice@Reading, University of Reading, UK

Abstract

In numerical weather prediction (NWP) PDEs are used to model the evolution of the atmosphere and ocean. The related optimisation problem of finding the most likely initial atmospheric state is extremely high dimensional, with $10^9$ state variables, and $10^7$ observations for each time window [1]. Additionally, the underlying models themselves are complex and expensive to evaluate. In practice this means that iterative methods are used, and that techniques such as preconditioning are vital in order to obtain a tractable problem.

One area of recent research interest has been the conditioning of data assimilation problems, which combine information about current observations with a simulated numerical model of the system (or background) in order to find the most likely initial state. This combination process is done via an objective function that measures the goodness of fit between the observations and background. In the case of variational data assimilation, which is equivalent to optimal state estimation, the objective function takes the form of a non-linear least-squares function. Contributions from observations and the background in the objective function are weighted according to their relative uncertainty via error covariance matrices. The use of correlated observation error covariance matrices is relatively recent [2, 5] and has raised problems with convergence of the optimisation process. The development of theory and application-driven methods for understanding the impact of changing individual aspects of the data assimilation system on convergence is therefore critical.

In this work we use the condition number of the Hessian of the objective function as a proxy to investigate the speed of convergence of the least squares minimisation in the presence of correlated observation error covariance matrices for the first time. We use numerical linear algebra results to develop new bounds on the condition number in terms of the constituent matrices of the Hessian for two formulations of the variational data assimilation problem.

We firstly consider the unpreconditioned variational problem as studied in [3]. The minimum eigenvalue of the observation error covariance matrix appears in the denominator of both upper and lower bounds, indicating that small eigenvalues of the OEC matrix are likely to result in a Hessian that is ill-conditioned. Numerical results in an idealised framework reveal that for different experimental cases, conditioning of the Hessian is dominated by either the background or observation error covariance matrix. The choice of observation network determined the smoothness of this transition between the behaviour. In our numerical framework, the conditioning of the Hessian represented the convergence of a conjugate gradient method well in many examples. For instances where the behaviour was different, repeated eigenvalues of the Hessian led to rapid convergence of the conjugate gradient method. This is a well-known case where the condition number provides a pessimistic upper bound on convergence.

We then consider the formulation that is typically solved at NWP centres, which makes use of a control variable transform in order to decorrelate the background term in the objective function.
This formulation was studied in [4]. This can be thought of as a form of preconditioning, and is known to be effective in the case that observation errors are uncorrelated. However, it is not known how well this choice of preconditioner will perform when correlated observation errors are introduced.

Using similar techniques to the unpreconditioned case, we find that the minimum eigenvalue of the OEC matrix appears in both upper and lower bounds, meaning that small eigenvalues of the OEC matrix are likely to lead to ill-conditioned Hessians. Numerical experiments revealed that, unlike in the unpreconditioned case, reducing the condition number of the background or observation error covariance matrix did not always decrease the condition number of the Hessian. This behaviour was not well-represented by our bounds, which separate the contribution of each term. For many cases, the condition number gives a good indication of how changes to the data assimilation problem are likely to affect convergence of a conjugate gradient method. However, there were also cases where clustered eigenvalues of the preconditioned Hessian led to much faster convergence than would be expected by simply considering its conditioning.

The qualitative conclusions from these theoretical studies of the variational data assimilation problem will help users understand how large changes to their system (such as the introduction of updated error statistics, or novel observation types) are likely to affect convergence of the minimisation. Indeed, theoretical knowledge about the contribution of each term to the conditioning of the Hessian will allow users to make more informed decisions when altering aspects of their data assimilation system.

References


Half-radial matrices and Crouzeix’s conjecture

IvetaHNétynková, Petr Tichý

Faculty of Mathematics and Physics, Charles University
www.karlin.mff.cuni.cz/~ptichy

Abstract

Numerical radius \( r(A) \) is the radius of the smallest ball with the center at the origin containing the field of values \( W(A) \) of a given square matrix \( A \). It is well-known that

\[
    r(A) \leq \| A \| \leq 2r(A),
\]

where \( \| \cdot \| \) denotes the matrix 2-norm. Matrices attaining the lower bound are called radial, and have been analyzed thoroughly. In this presentation we investigate matrices satisfying \( r(A) = \| A \| / 2 \), and call them half-radial. We summarize the existing results and formulate new ones. In particular, we investigate their singular value decomposition and algebraic structure. Based on that, we study the extreme case of the attainable constant 2 in Crouzeix’s inequality.

Half-radial matrices. It is well known that half-radial matrices are closely related to the \( 2 \times 2 \) Jordan block with the zero eigenvalue, and that their field of values is always a disk with the radius \( \| A \| / 2 \), centered at the origin. We will present several new necessary and sufficient conditions for a matrix \( A \) to be half-radial; for more details see [3, Theorem 9]. For example, we define a subset of the set of maximizers of \( |\langle Az, z \rangle| \) over the unit sphere,

\[
    \Theta_A \equiv \{ z \in \mathbb{C}^n : \| z \| = 1, \ r(A) = |\langle Az, z \rangle|, \ \langle Ax, x \rangle = 0 \},
\]

where \( z = x + y, \ x \in \mathcal{R}(A^*), \ y \in \mathcal{N}(A) \). Here \( \mathcal{R}(A^*) \) denotes the range of \( A^* \), \( \mathcal{N}(A) \) the null space of \( A \), and \( \langle \cdot, \cdot \rangle \) the Euclidean inner product. We show that

\[
    \| A \| = 2r(A) \iff \Theta_A \neq \{ \emptyset \}.
\]

This result supplements the known sufficient condition: If \( \mathcal{R}(A^*) \perp \mathcal{R}(A) \), then \( A \) is half-radial. We next study the structure of \( \Theta_A \). Our results [3, Lemma 6] imply that a half-radial matrix \( A \) has orthogonal maximum right and left singular subspaces. The maximum right and left singular vectors can be then used to characterize the structure of \( \Theta_A \). We prove that \( \Theta_A \) is either empty or

\[
    \Theta_A = \left\{ \frac{1}{\sqrt{2}} \left( e^{i\alpha}v + e^{i\beta}A\frac{v}{\| A \|} \right) : v \in \mathcal{V}_{\max}(A), \ \| v \| = 1, \ \alpha, \beta \in \mathbb{R} \right\},
\]

where \( \mathcal{V}_{\max}(A) \) is the maximum right singular subspace.

Algebraically, \( A \) is half-radial if and only if \( A \) is unitarily similar to the block diagonal matrix

\[
    (I_m \otimes J) \oplus B = \begin{bmatrix} J & \cdots & J \\ \vdots & \ddots & \vdots \\ J & \cdots & B \end{bmatrix}, \quad J = \frac{1}{\| A \|} \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix},
\]

where \( m = \dim \mathcal{V}_{\max}(A) \), and \( B \) satisfies \( \| B \| < \| A \|, \ r(B) \leq \frac{1}{2} \| A \| \); see [3, Lemma 8].
Crouzeix’s conjecture. Crouzeix’s conjecture [2] states that
\[ \|p(A)\| \leq 2 \max_{\zeta \in W(A)} |p(\zeta)| = 2 \| p \|_{W(A)} \]
holds for any square matrix \( A \) and any polynomial \( p \). Let us call the above inequality Crouzeix’s inequality. We investigate the set of matrices for which the bound in Crouzeix’s inequality can be attained, and prove the following result: For an integer \( k \geq 1 \), it holds that
\[ \|p(A)\| = 2 \| p \|_{W(A)} \]
for \( p(\zeta) = \zeta^k \) if and only if \( A^k \) is half-radial and \( r(A^k) = r(A)^k \).

The conditions that \( A^k \) is half-radial and \( r(A^k) = r(A)^k \) make the set of matrices very special. In more detail, Greenbaum and Overton conjecture in [4, p. 239] that if the equality holds in the above inequality for \( A \in \mathbb{C}^{(n+1)\times(n+1)} \) and the polynomial \( p(\zeta) = \zeta^n \), then \( A \) is unitarily similar to a scalar multiple of the \((n+1) \times (n+1)\) Crabb-Choi-Crouzeix matrix \( C_n \),

\[ C_1 = \begin{bmatrix} 0 & 2 \\ 0 & 0 \end{bmatrix}, \quad C_n = \begin{bmatrix} 0 & \sqrt{2} & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ldots & \vdots \\ 0 & 0 & \ldots & 0 & \sqrt{2} \\ 0 & 0 & \ldots & 0 & 0 \end{bmatrix}. \]

In other words, the authors of [4] conjecture that up to scaling and unitary similarity, there is just one matrix giving the equality for the polynomial \( p(\zeta) = \zeta^n \). Using results of Crabb [1] it is possible to show that their conjecture is true; see [3, Lemma 13]. Moreover, we present the structure of matrices satisfying the equality
\[ \|A^k\| = 2 \max_{\zeta \in W(A)} |\zeta^k| \]
for some \( 1 \leq k \leq n \). Such matrices are unitarily similar to a block diagonal matrix of the form
\[ r(A) \begin{bmatrix} C_k & B \\ \end{bmatrix}, \]
where \( r(B) \leq 1 \) and \( \|B^k\| \leq 2 \); see [3, Theorem 14].

Summary. Our results suggest that half-radial matrices are related to the case when the upper bound in Crouzeix’s inequality can be attained for some polynomial. This is also supported by another conjecture of Greenbaum and Overton [4, p. 242] predicting that the upper bound in Crouzeix’s inequality can be attained only if \( p \) is a monomial. In particular, if this conjecture is true, then the above result characterizes all matrices for which the upper bound in Crouzeix’s inequality can be attained.

References


Nonlinear eigenvalue problems arise in many areas of computational science and engineering, including acoustics, control theory, fluid mechanics, and structural engineering. The fundamental formulation of such problems is given in the following definition. Given a nonempty open set $\Omega \subseteq \mathbb{C}$ and a matrix-valued function $F : \Omega \to \mathbb{C}^{n \times n}$, the nonlinear eigenvalue problem (NEP) consists of finding scalars $\lambda \in \Omega$ (the eigenvalues) and nonzero vectors $v \in \mathbb{C}^n$ and $w \in \mathbb{C}^n$ (right and left eigenvectors) such that

$$F(\lambda)v = 0, \quad w^*F(\lambda) = 0^*.$$  

Clearly, the eigenvalues $\lambda$ of $F$ are the solutions of the scalar equation $\det F(\lambda) = 0$. An NEP can have no eigenvalue at all (e.g., if $F(\lambda) = \exp(\lambda)$), finitely many eigenvalues (e.g., if $F(\lambda) = \lambda^2M + \lambda D + K$ with $M, D, K \in \mathbb{C}^{n \times n}$), countably many (e.g., if $F(\lambda) = [e^{\lambda^2} 1 1]$), or a continuum of eigenvalues (e.g., $F(\lambda) = 0$). These make the computation of eigenvalues and eigenvectors difficult even for problems of small dimensions. In this talk we assume that $F$ is regular, i.e., $\det F(\lambda) \neq 0$, thereby excluding the situation where $F$ has a continuum of eigenvalues.

NEP solvers can be classified into three groups, those based on Newton’s method, solvers based on contour integrals, and solvers based on linearizations [4]. Newton’s method is a natural approach to compute a few eigenvalues or eigenpairs of NEPs very efficiently and accurately provided that very good initial guesses are available. The choice of an initial guess is typically the only crucial parameter of a Newton-type method, which is a great advantage over other NEP solution approaches.

Several Newton-type eigensolvers are available in the Julia package NEP-PACK [6]. Nevertheless these eigensolvers can be difficult to use as a black-box when no initial guesses are available.

NEP solvers based on linearizations are particularly well suited for large sparse NEPs and has been implemented in software such as NLEIGS [3], CORK [7], and in the NEP module of the SLEPc package [5].

Solvers based on contour integrals have also been used to solve very large problems due to their great potential for parallelization [1], [8]. They are equally well suited for small to medium size dense NEPs and can be used not only for holomorphic matrix functions $F$ but also for finitely meromorphic $F$ in $\Omega$, i.e., they can cope with singularities inside $\Omega$. Their implementation relies on parameters that are nontrivial to choose or set up for a given problem as we now explain.

Beyn’s integral approach [2] reduces an NEP with $m \leq n$ eigenvalues inside a contour $\Gamma$ to a $m \times m$ linear eigenproblem. It requires computing the matrices

$$A_0 := \frac{1}{2\pi i} \int_{\Gamma} F(z)^{-1}X \, dz, \quad A_1 := \frac{1}{2\pi i} \int_{\Gamma} z F(z)^{-1}X \, dz,$$

where $R \in \mathbb{C}^{n \times r}$ is a probing matrix whose size needs to be chosen (ideally $r$ should be the smallest integer $\leq n$ such that $A_0$ becomes singular). Then using $A_1$ and an economy-size SVD of $A_0$, $A_0 = V_0\Sigma_0 W_0^*$, a matrix $M := V_0^*A_1W_0\Sigma_0^{-1}$ is constructed whose eigenvalues are the eigenvalues of $F$ inside $\Gamma$.  

This approach only works when the number of eigenvalues inside the contour $\Gamma$ is less than the dimension $n$ of our matrix-valued function $F$. When $m > n$, $p$ higher order moments must be used to construct the block-Hankel matrices

$$
\begin{align*}
B_0^{[p]} &:= \begin{bmatrix}
A_0 & \cdots & A_{p-1} \\
\vdots & & \vdots \\
A_{p-1} & \cdots & A_{2p-2}
\end{bmatrix}, \\
B_1^{[p]} &:= \begin{bmatrix}
A_1 & \cdots & A_p \\
\vdots & & \vdots \\
A_p & \cdots & A_{2p-1}
\end{bmatrix}, \\
A_j &:= \frac{1}{2\pi i} \int_{\Gamma} z^j F(z)^{-1} X \, dz.
\end{align*}
$$

For $p$ and $r$ large enough (but not too large), $r$ being the number of columns of the probing matrix $X$, an economy-size SVD $B_0^{[p]} = V_0 \Sigma_0 W_0^*$ is used as before to construct $M := V_0^* B_1^{[p]} W_0 \Sigma_0^{-1}$ whose eigenvalues are the same as the eigenvalues of $F$ inside $\Gamma$ including partial multiplicities.

A quadrature approximation with $n_q$ quadrature nodes is used to compute the moments $A_j$ whose quality affects the accuracy of the computed eigenpairs.

For a given matrix-valued function $F$ and a contour $\Gamma$ around the eigenvalues of interest, our aim is to discuss how to systematically choose the parameters $p$ and $r$, and the number of quadrature nodes for an efficient and reliable implementation of a contour integral based eigensolver for NEPs.

References


Null-space approach for solving symmetric augmented systems

Jennifer Scott, Miroslav Tůma

Department of Numerical Mathematics, Faculty of Mathematics and Physics, Charles University
http://www.karlin.mff.cuni.cz/~mirektuma

Abstract

Our focus is on solving augmented systems of linear equations of the form

$$K \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} A_s^T A_s & A_s^T A_d \\ A_d^T & -C_d \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix},$$

(1)

using a null-space approach that is based on the null space basis of the block $A_d$. We assume that the system matrix $K$ is symmetric and indefinite and that the augmented system has a unique solution. We also assume that the row dimension of $A_d$ is much smaller than its column dimension.

It is straightforward to show that solving a system of the general form (1) is one way of solving the linear least squares problem

$$\min_x \|Ax - b\|_2,$$

(2)

where

$$A = \begin{pmatrix} A_s \\ A_d \end{pmatrix}, \quad A_s \in \mathbb{R}^{m_s \times n}, \quad A_d \in \mathbb{R}^{m_d \times n},$$

with $m = m_s + m_d$, $m_s \geq n$ and $n > m_d \geq 1$ is small ($m_d \ll m_s$). In this case, we assume that $A$ has full column rank but neither $A_s$ nor $A_d$ has to be a full rank matrix. An important example that arises in practice is $A_d$ comprising a block of dense rows. In certain situations, a null-space approach may be beneficial compared with other methods for solving (2). For example, the block of rows $A_d$ may represent constraints of the constrained least squares problem that have to be solved more accurately [2]. The null-space approach is one way to achieve this. Moreover, if $A_d$ changes, as happens when solving a sequence of problems in numerical optimization, the null-space approach may potentially be advantageous.

A solver for the system (1) based on the null space of the underdetermined block $A_d$ is straightforward when $C_d = 0$. The solution strategy is then well-known and is termed, for example, the dual variable or the reduced Hessian method. Solving the system when $C_d \neq 0$ is not straightforward; see, for example, the discussion in Benzi et al. in [1]. A one-sided approach that exploits a null-space strategy for (1) with $C_d \neq 0$ has been recently proposed by Howell [3] but this fails to preserve symmetry, trading a symmetric problem for a nonsymmetric one.

This talk proposes a new symmetry-preserving null-space approach. Assume that $\text{rank}(A_d) = r \leq m_d$ and that $C$ is symmetric positive definite (as we have, for example, in the least squares case). Further, suppose we have a nonsingular matrix $E = (Z \ Y) \in \mathbb{R}^{n \times n}$, where $Z \in \mathbb{R}^{n \times n-r}$ such that

$$A_d E = \begin{pmatrix} 0_{m_d, n-r} & A_r \end{pmatrix},$$

(3)

and $A_r \in \mathbb{R}^{m_d \times r}$ is nonsingular. We can then show that there exists $E \in \mathbb{R}^{(n+m_d) \times (n+m_d)}$ such that the transformed saddle point matrix

$$\tilde{K} = E^T K E$$

(4)
is symmetric and invertible having the symmetric positive definite leading principal submatrix $Z^T H Z$ of order $n - r$. The solution of the transformed saddle point system then leads directly to the solution of the original saddle point problem.

As the row dimension of $A_d$ is assumed to be much smaller than its column dimension, an inseparable ingredient of the null-space solver is the construction of the matrix $E$ that contains the null-space basis of a wide and possibly dense matrix. We propose approaches to perform this construction based on modifications of the turnback method originally developed to construct null-space bases of constraint sets in structural mechanics; see [5]. The principal part of the modification is to balance a high level of linear independence of the columns of $Z$ with the target sparsity of $Z^T A_s^T A_s Z$. This research is work in progress and is a new direction in our recent efforts to develop alternative ways to solve large-scale sparse-dense linear least squares problems by preconditioned iterative methods [4].

References


A Flexible Power Method for Infinite Dimensional Tensor Eigenvalue Problems

Roel Van Beeumen, Lana Periša, Daniel Kressner, and Chao Yang

Computational Research Division, Lawrence Berkeley National Laboratory

www.roelvanbeeumen.be

Abstract

We consider the problem of computing the leftmost, i.e., algebraically smallest, eigenvalue of the infinite dimensional Hermitian eigenvalue problem

\[ Hx = \lambda x, \]  

where the Hermitian matrix \( H \) is infinite dimensional and defined as the infinite sum of Kronecker products of an infinite number of finite matrices, i.e.,

\[ H = \sum_{k=-\infty}^{+\infty} \cdots \otimes I \otimes \otimes M_{k,k+1} \otimes I \otimes I \otimes \cdots, \]

with \( I \) the \( n \times n \) identity matrix and \( M_{k,k+1} \in \mathbb{C}^{n^2 \times n^2} \). We assume that the matrices \( M_{k,k+1} \) are identical for all \( k \), hence \( H \) is called to be translational invariant. The double subscript \( k, k+1 \) for the matrix \( M \) is used here to merely indicate the overlapping positions of \( M \) in each Kronecker product. This type of eigenvalue problems originates from the study of simple quantum many-body systems such as a quantum spin chain with nearest neighbor interactions.

The eigenvectors of \( H \) are infinite dimensional vectors which obviously cannot be computed or stored directly in memory. One way to study such a problem computationally is to start with a finite \( H_d \in \mathbb{C}^{n^d \times n^d} \) that only contains \( (d - 1) \) terms in the summation and Kronecker products of \( (d - 1) \) matrices, and examine how the smallest eigenvalue of \( H_d \) changes as \( d \) increases. Note that the Kronecker structure of \( H_d \) allows for efficiently representing this matrix in Tensor Train (TT) format [4]. The corresponding eigenvector can then be represented by the TT

\[ x_d(i_1, i_2, \ldots, i_d) = X_1(i_1)X_2(i_2)\cdots X_d(i_d), \]

where \( X_k(i_k) \) is an \( r_{k-1} \times r_k \) matrix, with \( r_0 = r_d = 1 \), and the indices \( i_k = 1, \ldots, n \), for \( k = 1, \ldots, d \). Because there is a limit on how large \( d \) can be chosen, we may never know the true solution in the limit \( d \to \infty \). Such a limit is known in the physics literature as the thermodynamic limit and is important for describing macroscopic properties of quantum materials when \( H \) corresponds to a quantum many-body Hamiltonian.

For directly computing the smallest eigenvalue of the infinite dimensional eigenvalue problem (1)–(2), we will represent the eigenvector in a compact form and make use of the translational invariance property of \( H \). Such invariance property was studied by Bethe [1] and Hulthén [2], known as the Bethe–Hulthén hypothesis, and suggests that the elements of the eigenvector are invariant with respect to a cyclic permutation of the tensor indices, i.e.,

\[ x(\ldots, i_{-1}, i_0, i_1, \ldots) = x(\ldots, i_0, i_1, i_2, \ldots), \]

where all indices \( i_k = 1, \ldots, n \), and \( \ldots, i_{-1}, i_0, i_1 \ldots \) is a sequence of indices that specify a particular element of the infinite dimensional vector \( x \).
We propose a flexible power method to compute the smallest eigenvalue of $H$ by representing the eigenvector as a translational invariant infinite Tensor Ring (iTR), defined as follows

$$x(\ldots, i_{-1}, i_0, i_1, \ldots) = \text{Tr} \left[ \prod_{k=-\infty}^{+\infty} X(i_k) \right],$$

where all indices $i_k = 1, \ldots, n$, and each $X(i_k)$ is a matrix of size $r \times r$. Note that due to the translational invariance, we only need to store, and work, with $n$ matrices of size $r \times r$, which is manageable as long as the rank $r$ is not too large. An iTR can be seen as the infinite limit of a finite size tensor ring [6] and is also known as a uniform Matrix Product State (uMPS) [5].

We assume that the smallest eigenvalue of $H$ is simple and apply a power method to $e^{-H}$ for computing this eigenvalue. In order to implement the power method, we must be able to multiply the matrix exponential $e^{-H}$ with an iTR and keep the product in iTR form. Computing the matrix exponential $e^{-H}$, with an infinite dimensional matrix $H$, applied to an iTR is in general not possible. However, the special structure of (2) allows us to split $H$ in even and odd terms, $H_e$ and $H_o$, respectively. Next, by introducing a small parameter $t$, we can use the Suzuki–Trotter splitting

$$e^{-Ht} \approx e^{-tH_o} e^{-tH_e},$$

to simplify the multiplication of $e^{-Ht}$ with an iTR into only local tensor contractions with the $n^2 \times n^2$ matrix exponential $e^{-Mt}$. Such contractions can be implemented so that the translational invariance is preserved, although the rank $r$ of the contracted tensor generally increases. To keep the cost of subsequent power iterations bounded, a low rank approximation through the use of a truncated singular value decomposition is applied. Finally, the smallest eigenvalue of $H$ can be approximated by the iTR Rayleigh quotient

$$\theta = \frac{\langle x, Hx \rangle}{\langle x, x \rangle}.$$  

We show how the contractions can be performed efficiently and how a truncated SVD is used to keep the rank of the iTR fixed. An iTR residual norm $\|Hx - \theta x\|$ is defined to monitor the convergence of the Rayleigh quotient and to modify the parameter $t$. We discuss 2 different implementations of the algorithm and prove that the iTEBD algorithm [3] converges to an iTR eigenvector in canonical form. We also illustrate the algorithm with several numerical examples.

References

References


Detecting roles in large graphs

Melissa Marchand, Kyle Gallivan, Wen Huang, and Paul Van Dooren

Department of Mathematical Engineering, Université catholique de Louvain, Belgium
https://perso.uclouvain.be/paul.vandooren/

Abstract

To analyze large networks and obtain relevant statistical properties, clustering nodes together into subgroups of densely connected nodes, called communities, is a popular approach. Various measures and algorithms have been developed to identify these community structures [6]. However, there are network structures that cannot be determined using community detection algorithms, such as overlapping community structures or bipartite and cyclic graph structures, which appear, for example, in human protein-protein interaction networks and food web networks, respectively. General types of network structures are known as role structures, and the process of finding them is called the role extraction problem, or block modeling.

The role extraction problem determines a representation of a network by a smaller structured graph, called the reduced graph, role graph, or image graph, where nodes are grouped together into roles based upon their interactions with nodes in either the same role or different roles. This problem is a generalization of the community detection problem where each node in a community mainly interacts with other nodes within the same community and there are no, or very few, interactions between communities. There are many real world applications to which role extraction can be applied and from which characterizations of interactions that define roles can be taken, such as studying trade networks between countries; evaluating the resilience of peer-to-peer networks; ranking web pages in search engines; studying human interaction by email correspondence; modeling protein-protein interactions; and analyzing food webs (see e.g. [8]).

Previous research solved the role extraction problem using either direct or indirect approaches, where direct approaches cluster the network directly into roles [5, 8, 7], while indirect approaches construct a similarity graph of the data set and then cluster highly similar roles together [2, 4, 1]. Both approaches have strengths and weakness for solving the role extraction problem. A strength of direct approaches is that it explicitly fits the data into a role structure. Unfortunately, there is no well-accepted measure to determine whether or not a role assignment fits the data, so a priori knowledge about the network is necessary or multiple role assignments must be tested to determine the best role structure for the data [5].

Indirect approaches do not require an assumption on the role assignment and may reveal complex network structures that are not apparent in the original data. The main problem with indirect approaches is that there exist several different types of graph similarity measures. In addition, many of these measure have been deemed unsuitable for the role extraction problem due to difficulties encountered when extracting role structures from certain types of graphs (e.g., regular graphs and normal graphs), loss of information (e.g., the origin, the destination, and the intermediate nodes involved in the transmission of the flow), or were more suited to detect community structures than role structures [3].

Recently, Browet and Van Dooren introduced a new neighborhood pattern similarity measure [3]. They showed empirically that the measure was able to recover the role structure of complex networks and they developed an algorithm to compute a low-rank approximation of this similarity matrix. In this presentation, we analyze the neighborhood pattern similarity measure and show...
that for the “ideal” case of regular equivalent graphs, we can recover roles exactly from a low-rank factorization of the similarity matrix due to the relationship between the rank of the similarity matrix and the number of roles. We then unify special complex structures in networks, e.g., overlapping communities and bi-partite communities, that can be expressed as roles structures and identified using the neighborhood pattern similarity measure.

Since the role extraction problem is a combinatorial optimization problem, its complexity can be prohibitive. We show that the low rank iterative scheme to approximate the similarity measure for very large networks allows us to project our similarity measure onto a low rank manifold (say, of rank $r$), and that the role models can be recovered using classical clustering techniques in a $r$-dimensional space, which reduces the complexity significantly. We then show that our low rank similarity score successfully extracts the different roles in random graphs of Erdős-Rényi type. Finally, we illustrate these ideas with a number of real-world examples.

References


The influence of condition numbers on the convergence of rank-constrained optimization problems

André Uschmajew and Bart Vandereycken

Section of Mathematics, University of Geneva
https://www.unige.ch/math/vandereycken/

Abstract

Let $f: \mathbb{R}^{m \times n} \rightarrow \mathbb{R}$ be a strongly convex function. We consider the rank-constrained optimization problem

$$\min_{X \in \mathbb{R}^{m \times n}} f(X) \text{ subject to } \operatorname{rank}(X) \leq k. \quad (1)$$

Equivalently, we want to minimize $f$ over $\mathcal{M}_{\leq k}$, the set of $m \times n$ matrices with rank bounded by $k$. Problem (1) occurs in the approximate solution of certain matrix equations, low-rank matrix completion and sensing, and phase retrieval. Generalizing matrices to tensors, such problems are also used in theoretical physics for the computation of the ground state of the Schrödinger equation. The set $\mathcal{M}_{\leq k}$ then contains tensors with bounded TT rank. See [4] for a recent overview.

Problem (1) is non-convex in general due to the rank constraint $\mathcal{M}_{\leq k}$. It is therefore surprising that algorithms that use only local information (like the gradient of $f$) can be used to solve (1). A good example of such a method is block coordinate descent: since a rank $k$ matrix can be factored as $X = LR^T$, we perform

$$L_{k+1} = \arg \min_{L \in \mathbb{R}^{m \times k}} f(L R_k^T),$$
$$R_{k+1} = \arg \min_{R \in \mathbb{R}^{n \times k}} f(L_{k+1} R^T).$$

This method is especially attractive for quadratic $f$ since then the subproblems are solved by least-squares. Other algorithms for solving (1) include methods based on Riemannian optimization since the set of matrices of fixed rank is a smooth manifold.

In this talk, I will explain how the condition number of $X$ and of $\nabla^2 f(X)$ influences the local and global convergence of some of these methods.

**Local convergence.** Like in the Euclidean case, the local convergence of Riemannian optimization methods is determined by the spectrum of the Riemannian Hessian of $f$ at the limit point $X_*$; see [1]. In case where $f$ is a zero residual problem, this leads to asymptotic rates that do not depend on $\kappa(X_*)$, the condition number of $X_*$. This is in sharp contrast when optimizing $f(L R^T)$ jointly over $L$ and $R$ since then the convergence slows down for increasing $\kappa(X_*)$.

On the other hand, the good performance of the block coordinate descent methods can be understood if we analyze the iteration in terms of subspaces. Instead of optimizing over $L$ or $R$, it selects elements from the subspace

$$T_1(X_k) = \{LR_k^T : L \in \mathbb{R}^{m \times k}\} \quad \text{or} \quad T_2(X_k) = \{L_k R^T : R \in \mathbb{R}^{n \times k}\}.$$

The condition number of the iterate $X_k$ has now no immediate effect on the Lipschitz constant of the objective function $f$ restricted to $T_1$ or $T_2$. This leads to convergence factors that do not depend on $\kappa(X_*)$, regardless of $f$.

**Global convergence.** Let $f(X) = \frac{1}{2} \langle X, A(X) \rangle + \langle X, B \rangle$ be quadratic such that $A$ is positive definite with condition number $\kappa(f)$. When $\kappa(f) \lesssim 3$, one can show that the optimization landscape
is benign in the sense that (a) all local minima are close to the global minimum and (b) all other critical points are saddle points with sufficient negative curvature. This property guarantees that steepest descent with random perturbations provably converges to a neighbourhood of the global minimum while avoiding the saddle points; see [2].

There exist counterexamples of the benign landscape when $\kappa(f) > 3$, but it is an open problem to explain the good convergence for more relevant operators $\mathcal{A}$, like discretized Laplacians. A slight generalization is possible for operators that satisfy the following restricted positive definite property (RPD):

$$
\exists \delta_{2k} < 0.28 \text{ such that } (1 - \delta_{2k}) \|X\|^2_F \leq \|\mathcal{A}(X)\|^2_F \leq (1 + \delta_{2k}) \|X\|^2_F \quad \forall X \in \mathcal{M}_{\leq 2k}.
$$

Even though $\mathcal{A}$ does not need to be invertible on $\mathbb{R}^{m \times n}$, it still behaves like a perturbation of identity on $\mathcal{M}_{\leq 2k}$ since its restricted singular values are bounded by $1 - \delta_{2k}$ and $1 + \delta_{2k}$; see [3].

References


Low-rank plus Sparse Matrices: Ill-posedness and Recovery

Jared Tanner and Simon Vary

Mathematical Institute, University of Oxford
https://people.maths.ox.ac.uk/vary/

Abstract

Low-rank matrix models are ubiquitous in scientific computing and statistics. They offer statistical interpretations (Principal Component Analysis, Linear Discriminant Analysis), lower sample complexity (matrix sensing [2]), and lower computational and space complexity (factorized semidefinite programming [5], model reduction).

Herein, we study the question of what happens in the case when the low-rank matrix assumption does not hold exactly. That is, we work with matrices $M \in \mathbb{R}^{m \times n}$ of the form

$$M = L + S,$$

(1)

where $L$ is of rank at most $r$, and $S$ has at most $s$ non-zero entries, $\text{nnz}(S) \leq s$. The purpose of $S$ is to represent the model outliers or distribution-independent noise that is limited in quantity. Define the following set of low-rank plus sparse matrices

$$\mathcal{LS}_{m,n}(r,s) = \{ L + S \in \mathbb{R}^{m \times n} : \text{rank}(L) \leq r, \text{nnz}(S) \leq s \}.$$  

(2)

We study lower semicontinuity of the matrix rigidity function

$$\text{Rig}(M, r) = \min \{ \text{nnz}(S) : \text{rank}(M - S) \leq r \},$$

(3)

originally introduced in [1] for the purpose of deriving lower bounds for the computational complexity of linear transforms. By constructing examples of matrices for which the matrix rigidity function is not lower semicontinuous, we also prove that the $\mathcal{LS}_{m,n}(r,s)$ sets are not closed for a wide range of ranks, sparsities and matrix sizes.

**Theorem 3** ($\mathcal{LS}_{m,n}(r,s)$ is not closed [6]) The set of low-rank plus sparse matrices $\mathcal{LS}_{m,n}(r,s)$ is not closed for $r \geq 1$, $s \geq 1$ provided $(r+1)(s+2) \leq n$, or provided $(r+2)^{3/2}s^{1/2} \leq n$ where $s$ is of the form $s = p^2 r$ for an integer $p \geq 1$.

As a consequence, we demonstrate that many non-convex optimization problems over the set of low-rank and low-rank plus sparse matrices (e.g. robust PCA, low-rank matrix completion, factorized semidefinite programming) can be ill-posed in the sense that they can have no global minimum.

If one restricts the norm of one of the components, the low-rank plus sparse set is formed as an algebraic sum of a closed set and a compact set, thus making the following set closed

$$\mathcal{LS}_{m,n}(r,s,\tau) = \{ L + S \in \mathbb{R}^{m \times n} : \text{rank}(L) \leq r, \text{nnz}(S) \leq s, \|L\|_F \leq \tau \}.$$  

(4)

Restricting the norm alleviates the issue of an open boundary, guaranteeing that optimization over $\mathcal{LS}(r,s,\tau)$ is well-posed.

We consider the problem of finding a low-rank plus sparse matrix $L_0 + S_0 = M \in \mathcal{LS}_{m,n}(r,s,\tau)$ that satisfies a given system of linear constraints $\mathcal{A}(M) = b$, where $\mathcal{A} : \mathbb{R}^{m \times n} \to \mathbb{R}^p$, $b \in \mathbb{R}^p$ and typically $p \ll mn$. 
It is well-known that such problems are NP-hard in general. In this work we show that if the linear transformation $\mathcal{A}$ satisfies the restricted isometry property (RIP) for $\text{LS}_{m,n}(r,s,\tau)$ sets, the solution can be recovered either by solving a convex optimization problem

$$\min_{X=L+S \in \mathbb{R}^{m \times n}} \|L\|_* + \lambda \|S\|_1, \quad \text{s.t.} \quad \mathcal{A}(X) = b, \quad (5)$$

where $\| \cdot \|_*$ is the Schatten 1-norm and $\| \cdot \|_1$ is the $\ell_1$-norm, or by iterative methods that we prove to be guaranteed to converge to a global minimum of the non-convex optimization problem

$$\min_{X=L+S \in \mathbb{R}^{m \times n}} \|\mathcal{A}(X) - b\|_F, \quad \text{s.t.} \quad X \in \text{LS}_{m,n}(r,s,\tau). \quad (6)$$

Additionally, we show that for random ensembles of linear transformations that satisfy certain concentration of measure inequalities, the RIP for $\text{LS}_{m,n}(r,s,\tau)$ holds asymptotically as $m,n,p$ approach infinity at appropriate rates. In particular, it is necessary for $p$ to be taken proportional to the order of degrees of freedom of the rank-$r$ plus sparsity-$s$ matrices times a logarithmic factor.

**Theorem 4 (RIP for $\text{LS}_{m,n}(r,s,\tau)$)** For a given $m,n,p$ and $\delta \in (0,1)$ and a random linear transform $\mathcal{A} : \mathbb{R}^{m \times n} \to \mathbb{R}^p$ satisfying the concentration of measure inequalities, there exist constants $c_0,c_1 > 0$ such that the RIP for $\text{LS}_{m,n}(r,s,\tau)$ holds with $\delta_{r,s,\tau}(\mathcal{A}) \leq \delta$ provided

$$p > c_0 \left( r(m + n) + s \right) \log \left( \frac{1}{\tau} \frac{mn}{s} \right),$$

with probability at least $1 - \exp(-c_1p)$, where $c_0,c_1$ depend only on $\delta$.

Examples of random ensembles of $\mathcal{A}$ which satisfy the conditions of Theorem 4 include random Gaussian ensembles, symmetric Bernoulli ensembles, and Fast Johnson-Lindenstrauss Transform (FJLT) [4].

Consequently, we prove that the solution $X^* = L^* + S^*$ of the convex optimization problem formulated in (5) recovers the matrix $M$.

**Theorem 5 (Guaranteed convex recovery)** Suppose that $r,s \geq 1$ and $\tau > 0$ are such that the restricted isometry constant $\delta_{r,s,\tau}(\mathcal{A}) \leq \frac{\tau}{s}$, and let $X^* = L^* + S^*$ be the solution of (5) with $\lambda = \sqrt{r/s}$. Then $L^* = L_0$, $S^* = S_0$, and $X^* = M$.

We propose two computationally efficient iterative methods called Normalized Iterative Hard Thresholding (NIHT) [3] and Normalized Alternative Hard Thresholding (NAHT) for low-rank plus sparse matrices that provably converge to a global minimum of (6).

**References**


Overlapping Community Detection via Eigendecomposition

Jimit Majmudar, Stephen Vavasis

Combinatorics & Optimization, University of Waterloo
https://uwaterloo.ca/scholar/vavasis

Abstract

The community detection problem is the following. There is a universe of \( n \) individuals, and there are \( k \) subsets \( C_1, \ldots, C_k \) of \( \{1, \ldots, n\} \) called “communities,” not known a priori. One is given a symmetric \( n \times n \) matrix \( A \) that indicates interactions among the individuals, and the problem is to identify \( C_1, \ldots, C_k \). This problem arises in many applications including analysis of social networks, analysis of authorship patterns in a scientific area, and activation of neurons in the brain. The most intensively studied version of this problem restricts attention to the case that \( C_1, \ldots, C_k \) are pairwise disjoint, which generalizes classic problems in data science such as clustering and the dense-subgraph problem. Herein we consider the community detection problem in which the communities may overlap, which is more difficult because of issues with uniqueness of the community decomposition.

Consider the following model for how \( A \) arises from \( C_1, \ldots, C_k \), which is perhaps the simplest possible: \( A(i,j) \) is equal to the number of communities in common between nodes \( i \) and \( j \). In this case, after an appropriate definition of diagonal entries of \( A \), the rank of \( A \) is at most \( k \), and in fact, \( A \) may be written as \( A = VV^T \) where \( V \) is an \( n \times k \) matrix of 0s and 1s such that \( V(i,j) \) indicates whether individual \( i \) is a member of community \( j \). It should be observed that even in this simple and noiseless model, an eigendecomposition of \( A \) does not recover \( V \) since the columns of \( V \) are not orthogonal; in fact, the eigenvectors of \( A \) will have entries of mixed signs, so it is not obvious how to recover communities. An assumption that guarantees recoverability in the noiseless case is the “pure node” assumption: each community \( C_j, j = 1, \ldots, k \), has at least one node \( i \in C_j \) such that \( i \) does not belong to any other community.

Our approach consists of two phases: first, find the eigendecomposition of \( A \), and second, solve convex relaxations involving the invariant subspaces. We show that the approach is guaranteed to recover communities under a reasonable model of noise in the data and under an extended pure-node assumption.
Algebraic and Optimization-based Algorithms for Decomposing Tensors into Block Terms

Nico Vervliet, Ignat Domanov, Lieven De Lathauwer

Department of Electrical Engineering (ESAT)—Stadius, KU Leuven
https://homes.esat.kuleuven.be/~nvervlie

Abstract

Tensors, or multiway arrays of numerical values, and their decompositions have been applied successfully in a myriad of applications in, a.o., signal processing, data analysis and machine learning [1]. Key is the ability to use decompositions to extract simple components under mild uniqueness conditions. The simplest decomposition, the canonical polyadic decomposition (CPD), writes a tensor as a sum of rank-1 tensors. However, in some applications more complex terms are required and a block term decomposition (BTD) may be more suitable.

In this work, we consider the decomposition of a third-order $I \times J \times K$ tensor $T$ into a sum of multilinear rank $(M_r, N_r, \cdot)$ terms:

$$T = \sum_{r=1}^{R} D_r \cdot_1 A_r \cdot_2 B_r \tag{1}$$

in which $D_r$, $A_r$ and $B_r$ have dimensions $M_r \times N_r \times K$, $I \times M_r$ and $J \times N_r$, resp., $\cdot_n$ denotes the mode-$n$ product, i.e., $T = D \cdot_n A \Leftrightarrow T_{(n)} = AD_{(n)}$, and the subscript $(n)$ denotes the mode-$n$ unfolding; see [1]. Alternatively, one can see this as the joint block diagonalization of the frontal slices $T_k$, $k = 1, \ldots, K$:

$$T_k = A \cdot \text{blockdiag}(D_{1,k}, \ldots, D_{R,k}) \cdot B^T, \tag{2}$$

in which $A = [A_1, \ldots, A_R]$ and $B = [B_1, \ldots, B_R]$. This problem has been studied for some special cases; see, e.g., [2, 3]. We tackle the general BTD, which is often solved via a nonconvex optimization problem which has many local optima. For the CPD and the decomposition into multilinear rank $(L_r, L_r, 1)$ terms (LL1), algebraic methods often provide suitable initializations. Here, we aim to derive an algebraic and an optimization-based method for problem (1).

**Algebraic algorithm** We show that the decomposition (1) can be cast as a coupled simultaneous eigenvalue decomposition (CS-EVD) problem under mild assumptions. Let $T \in \mathbb{K}^{I \times J \times K}$ ($\mathbb{K}$ can be $\mathbb{R}$ or $\mathbb{C}$) admit a decomposition into a sum of $R$ multilinear rank $(M_r, N_r, \cdot)$ terms (Eq. (1)) and assume $\sum_{r=1}^{R} M_r \leq I$ and $\sum_{r=1}^{R} N_r \leq J$. (Using compression by multilinear singular value decomposition [4], we can assume w.l.o.g. that these are equalities.) Let $A$ and $B$ have full column rank. Then, under some mild conditions, the decomposition in Eq. (1) is unique (see [5]), and can be reduced to a CS-EVD as follows. Let $M \in \mathbb{K}^{IJK \times (I^2 + J^2)}$ be defined as

$$M = \begin{bmatrix}
T_1^T \otimes I_I & -I_J \otimes T_1 \\
\vdots & \vdots \\
T_K^T \otimes I_I & -I_J \otimes T_K
\end{bmatrix}. \tag{3}$$

We can show that null$(M) =: N$ has dimension $R$ and can be partitioned as $[X \ Y]$ in which $X \in \mathbb{K}^{I^2 \times R}$ and $Y \in \mathbb{K}^{J^2 \times R}$. Let $X_r \in \mathbb{K}^{I \times I}$ ($Y_r \in \mathbb{K}^{J \times J}$) be the reshaped $r$th column $x_r$ ($y_r$),
\[ r = 1, \ldots, R, \text{ of } X (Y). \text{ We can show that } A \text{ and } B \text{ can be recovered from the CS-EVD problem} \]

\[ X_r = A \cdot \text{blockdiag}(g_{1r}I_{M_1}, \ldots, g_{Rr}I_{M_R}) \cdot A^{-1}, \quad r = 1, \ldots, R, \quad (4) \]

\[ Y_r = B^{-T} \cdot \text{blockdiag}(g_{1r}I_{N_1}, \ldots, g_{Rr}I_{N_R}) \cdot B^{-T}, \quad r = 1, \ldots, R. \quad (5) \]

Hence, we have two sets of simultaneous EVD problems that are coupled through the eigenvalues \( g_{rs}, r = 1, \ldots, R, s = 1, \ldots, R, \) which result in the factors \( A \) and \( B \) in Eq. (2). The core tensors \( D_r \) can be recovered from Eq. (2) as \( A \) and \( B \) are invertible by assumption. Note that \( R, M_r \) and \( N_r \) can be determined automatically from the dimension of the null space and the eigenvalues \( g_{rs}. \)

To reduce the computational cost of computing the null space \( N \) of the \( IJK \times (I^2+J^2) \) matrix \( M \), an iterative EVD algorithm is used to compute \( N \) as the subspace corresponding to the zero eigenvalues of \( M^H M. \) The bottleneck computation is solving \( M^H Mx = y \) for \( x \) every iteration. We show that the QR factorization of \( M = QR \) can be computed efficiently by exploiting the Kronecker and block structures using ideas from the block-QR and quasi-QR factorizations [6]. This way, \( M^H Mx = y \) can be written as \( R^H Rx = y \) which can be solved using forward and backward substitution as \( R \) is upper triangular. The cost of computing \( N \) then is \( O(J^4 + I^2J^2 + I^3) \) flop per iteration for the EVD step and \( O(J^6 + I^4K + I^2J^3K) \) flop for the QR step.

**Optimization-based algorithm** The CS-EVD problem (Eqs. (4)–(5)) is a coupled LL1 decomposition: by stacking \( X_r \) and \( Y_r \) as frontal slices in \( X \) and \( Y \), resp., we have (using CPD notation):

\[ X = [A, A^{-T}, GP_1], \quad Y = [B^{-T}, B, GP_2], \quad (6) \]

in which \( G \) collects the eigenvalues in Eqs. (4)–(5) and \( P_1 = \text{blockdiag}(1_{M_1}^T, \ldots, 1_{M_R}^T) \) and \( P_2 = \text{blockdiag}(1_{N_1}^T, \ldots, 1_{N_R}^T). \) We propose a single-step optimization-based approach that exploits this and computes the null space and the decomposition simultaneously by solving

\[ \min_{A,B,G} \frac{1}{2} \sum_{k=1}^{K} \|[A, T_k^T A^{-T}, GP_1] - [T_k B^{-T}, B, GP_2]\|_F^2. \quad (7) \]

This single-step approach has several advantages: increased robustness to noise, reduced accumulation of numerical error, and the possibility to exploit structure in both \( M \) and \( N \) simultaneously. The problem can be solved efficiently using Gauss–Newton in combination with standard CPD techniques and parametric constraints; see [7]. To avoid zero solutions, we can set \( G = I. \)

**References**


In mathematical control theory, the attenuation of noise and the treatment of uncertainties in a dynamical system plays a prominent role. Let this dynamics be described by the plant model \( P \)
\[
\frac{d}{dt} x(t) = Ax(t) + B_1 w(t) + B_2 u(t), \quad x(0) = x^0,
\]
\[
z(t) = C_1 x(t) + D_{11} w(t) + D_{12} u(t),
\]
\[
y(t) = C_2 x(t) + D_{21} w(t) + D_{22} u(t),
\]
where \( A \in \mathbb{R}^{n \times n} \), \( B_j \in \mathbb{R}^{n \times m_j} \), \( C_i \in \mathbb{R}^{n \times n} \), and \( D_{ij} \in \mathbb{R}^{p_i \times m_j} \) for \( i, j = 1, 2 \). Further, \( x : [0, \infty) \rightarrow \mathbb{R}^n \) is the state of the system, \( w : [0, \infty) \rightarrow \mathbb{R}^{m_1} \) is a noise input, \( u : [0, \infty) \rightarrow \mathbb{R}^{m_2} \) is the control input, \( z : [0, \infty) \rightarrow \mathbb{R}^{p_1} \) is the performance output, and \( y : [0, \infty) \rightarrow \mathbb{R}^{p_2} \) is the measured output.

In order to achieve desired performance and robustness properties of the dynamics, the given dynamical system is usually coupled with a so-called \( H_\infty \)-controller \( K \) that has to be designed in an optimal way. Such a controller is typically given by
\[
\frac{d}{dt} x_K(t) = A_K x_K(t) + B_K y(t), \quad x_K(0) = x^0_K,
\]
\[
u(t) = C_K x_K(t) + D_K y(t),
\]
where \( A_K \in \mathbb{R}^{n_K \times n_K} \), \( B_K \in \mathbb{R}^{n_K \times p_2} \), \( C_K \in \mathbb{R}^{m_2 \times n_K} \), \( D \in \mathbb{R}^{m_2 \times p_2} \), and \( x_K : [0, \infty) \rightarrow \mathbb{R}^{n_K} \) is the internal controller state.

The goal in \( H_\infty \)-control consists of choosing the matrices \( A_K, B_K, C_K \), and \( D_K \) such that the closed-loop system obtained by combining \( P \) and \( K \) is internally stable and the \( H_\infty \)-norm of the closed-loop transfer function is minimized. More precisely, if \( \mathcal{P}(s) = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \) is the transfer function of \( P \) (with the same block structure as \( P \)) and \( K(s) \) is the transfer function of \( K \), then the closed-loop transfer function is
\[
\mathcal{G}(K)(s) = \mathcal{P}_{11}(s) + \mathcal{P}_{12}(s) K(s)(I - \mathcal{P}_{22}(s) K(s))^{-1} \mathcal{P}_{21}(s).
\]

Then the \( H_\infty \)-control problem is
\[
\min_{K \text{ stabilizes } P} \| \mathcal{G}(K) \|_{H_\infty} := \min_{K \text{ stabilizes } P} \max_{\omega \in \mathbb{R} \cup \{-\infty, \infty\}} \| \mathcal{G}(K)(i\omega) \|_2.
\]

This is a very challenging constrained nonlinear, nonconvex, and nonsmooth optimization problem. Classical and well-established methods [1] address this problem by solving a sequence of algebraic Riccati inequalities or Riccati equations from which explicit formulas for the optimal controller \( K \) can be obtained. However, this methodology has two major drawbacks:

- Solving algebraic Riccati inequalities or equations can be quite challenging, in particular, if the state-space dimension \( n \) is large. Moreover, numerical problems may appear in the vicinity of the optimal performance level.
• Typically, the computed controller has the same order as the plant, i.e., $n = n_K$. Hence, such controllers are of low practical use unless a controller reduction is performed afterwards.

Therefore, the goal of this talk is to present methods that compute a $\mathcal{H}_\infty$-controller by direct optimization methods such as [2]. We assume that the plant dynamics is high-dimensional with large $n$, but the controller dynamics is low-dimensional with $n_K \ll n$. To use gradient-based optimization techniques, efficient methods to compute the $\mathcal{H}_\infty$-norm and its gradient with respect to the controller variables are needed for the large-scale case. In this talk we present a new method [3] to achieve this goal. For a given transfer function $G(s) = C(sI_n - A)^{-1}B + D$, the method generates a sequence of reduced transfer functions $\tilde{G}_1(s), \tilde{G}_2(s), \ldots$ whose $L_\infty$-norms can be computed very efficiently and with $\lim_{k \to \infty} \|G_k\|_{L_\infty} = \|G\|_{\mathcal{H}_\infty}$. The reduced transfer functions are created by a Petrov-Galerkin projection, i.e., $\tilde{G}_k(s) = C_k(s\tilde{E}_k - \tilde{A}_k)^{-1}\tilde{B}_k + \tilde{D}_k$, where

$$\tilde{E}_k = W_k^HV_k, \quad \tilde{A}_k = W_k^HAV_k, \quad \tilde{B}_k = W_k^HB, \quad \tilde{C}_k = CV_k, \quad \tilde{D}_k = D,$$

and $V_k, W_k$ have only a small number of columns. The projection matrices $V_k, W_k$ are constructed such that certain Hermite interpolation properties between $G(s)$ and $\tilde{G}_k(s)$ are satisfied, see [4]. In each step, the $\mathcal{L}_\infty$-norm of the reduced function is evaluated and the projection spaces are updated accordingly. This together with the Hermite interpolation properties gives a superlinear convergence result. Instead of projections it is also possible to compute the $\mathcal{H}_\infty$-norm by using the Loewner framework [5]. There, the reduced transfer functions are constructed with the help of so-called Loewner matrices that only need transfer function evaluations to be formed. This strategy is particularly useful, since in many applications the transfer functions under consideration are given in product or LFT form such as (1) or may not even be rational (such as for delay systems). We will also address the problem of convergence to only a local optimum of $\max_{\omega \in \mathbb{R} \cup \{-\infty, \infty\}} \|G(K)(i\omega)\|_2$.

We present a global optimality certificate that is based on a sequence of structure-preserving shift-and-invert Arnoldi iterations on a related even eigenvalue problem.

With these results at hand, it is now possible to deal with the $\mathcal{H}_\infty$-control problem. There it is possible to exploit the LFT structure of the closed-loop transfer function (1) which can be interpreted as a low-rank update of the plant model $P$ with the controller $K$. Moreover, if time permits, we show how these results can be exploited to significantly improve the classical D/K iteration in robust control which is needed if further model uncertainties are present.

References


---

1Here we need the $\mathcal{L}_\infty$-norm, since $\tilde{G}_k(s)$ may be unstable, whereas the $\mathcal{H}_\infty$-norm is only defined for stable transfer functions.


Preconditioning for nonsymmetric problems

Andy Wathen

Mathematical Institute, Oxford University, UK
www.maths.ox.ac.uk/people/andy.wathen

Abstract

Whilst we have an inadequate understanding of the convergence of Krylov subspace methods for nonsymmetric matrix systems, preconditioning remains largely heuristic for most nonsymmetric problems except those for which Elman’s classic GMRES(1) bound gives useful information. It is nevertheless widely appreciated from practical experience how valuable preconditioning can be in making feasible many large scale computations. Here we will pull together several useful theoretical observations and make some comments on possible approaches.
Solving the Matrix Eigenvalue Problem: chasing bulges, cores, or poles

David S. Watkins
Department of Mathematics, Washington State University
http://www.math.wsu.edu/faculty/watkins/

Abstract

Sixty years ago John Francis invented the winning general-purpose procedure for computing eigenvalues of a matrix, the implicitly-shifted QR algorithm. Variants of this algorithm can also be applied to related problems, including the generalized eigenvalue problem $Ax = \lambda Bx$. One might think that after so many years, everything that can be said on this topic has already been said, but this turns out not to be the case. A recent burst of activity has resulted in interesting new variants of and insights into Francis’s algorithm. It is normally implemented as a bulge-chasing algorithm, but recently we have shown that there are some advantages to implementing it by core chasing instead. This formulation leads to fast algorithm for unitary, unitary-plus-rank-one, and unitary-plus-low-rank matrices, including companion matrices. Another new and interesting variant, which is most naturally applied to the generalized eigenvalue problem, is the pole-swapping algorithm of Camps, Meerbergen, and Vandebril. This arose from considerations related to Ruhe’s rational Arnoldi process. We will describe, compare, and contrast the bulge-chasing, core-chasing, and pole-swapping approaches to solving the eigenvalue problem.

Much of the material of this talk is from joint work with others, including Jared Aurentz, Daan Camps, Thomas Mach, Leonardo Robol, and Raf Vandebril.
Fast randomized iterative numerical linear algebra and application to quantum chemistry

Timothy Berkelbach, Samuel Greene, Lek-Heng Lim, Jonathan Weare, and Robert Webber

Courant Institute of Mathematical Sciences
cims.nyu.edu/~weare

Abstract

Drawing upon ideas from the highly successful diffusion Monte Carlo (DMC) technique used in contexts ranging from electronic structure calculations to rare event simulation and data assimilation, we propose a new class of randomized iterative algorithms that we collectively call Fast Randomized Iteration schemes. DMC is primarily used to efficiently approximate expectations of the type appearing in Feynman–Kac formulae, i.e., for weighted expectations of Markov processes typically associated with parabolic partial differential equations. Our algorithms are based on similar principles but address a variety of common tasks in numerical linear algebra: linear systems, eigenvector problems, and matrix exponentiation, i.e., solving for $v$ in

$$Av = b, \quad Av = \lambda v, \quad v = \exp(A)b$$

for arbitrary matrices $A \in \mathbb{C}^{n \times n}$ that might not have any natural association with a Markov process (in particular, the entries of $A$ need not be nonnegative reals).

From the point of view of numerical linear algebra, the main novelty of the Fast Randomized Iteration schemes (FRI) is that they work in either linear or constant cost per iteration (and in total, under appropriate conditions) — they are intended for situations where $n$ is far beyond the reach of current technology. While traditional iterative methods in numerical linear algebra were created in part to deal with instances where a matrix (of size $O(n^2)$) is too big to store, the algorithms that we propose can be effective even in instances where the solution vector itself (of size $O(n)$) may be too big to store or manipulate. Our work is motivated by recent DMC based quantum Monte Carlo schemes that have been applied to matrices as large as $10^{108} \times 10^{108}$.

At each iteration of an FRI scheme, the matrix-vector product at the core of the underlying classical iterative numerical linear algebra technique is compressed, resulting in a sparse random vector. This sparse vector is an unbiased approximation of the vector it replaces. For example, we might replace the classical power iteration by

$$V_{t+1} = \frac{A(V_t + \eta(V_t))}{\|A(V_t + \eta(V_t))\|_1}$$

where, for any input vector $v$, $\eta(v)$ is a random perturbation chosen so that

$$\|v + \eta(v)\|_0 \leq m \quad \text{and} \quad \mathbb{E}[\eta(v)] = 0$$

for some user chosen integer $m$. The realizations of $\eta(V_t)$ are independent at each iteration. The cost of the matrix-vector multiplication is now reduced to $O(nm)$. The cost of the compression operation itself is generally $O(n)$. If the matrix $A$ is sparse, or if we introduce additional compression of $A$, the total cost of the matrix-vector multiplication and compression operations can be reduced to $O(m)$. Care must be taken in designing the random compression step as the performance of FRI is sensitive to its variance. Straightforward approaches, like the random entry selection used in other randomized numerical linear algebra techniques, result in unstable algorithms. I will discuss some of the essential features of effective compression rules.
Under ideal conditions the cost to achieve fixed accuracy with FRI schemes can be independent of the size of the target matrix. Here the choice of error norm is important. In high dimensions, the random perturbations introduced in each iteration will not be small if measured in the $\ell_1$. Low dimensional projections of those perturbations are, however, small. Our goal is therefore only to produce a random vector $V$ whose dot products with a small number of predetermined vectors are close to the dot products of those vectors with the solution vector $v$. Our chosen notion of error

$$\sup_{f \in \mathbb{C}^n} \sqrt{\mathbb{E} \left[ f \cdot (V - v) \right]}$$

reflects this goal. We leverage the stability of the underlying classical scheme to show that the accumulated affect of the perturbations introduced at each iteration can be very small when measured in this norm even when $n$ is extremely large. For most problems one still expects some dependence of the error on $n$. Though FRI and related schemes are observed to perform well in extremely high dimensions, specifying that dependence mathematically is challenging ongoing work.

Finally, I will report on our ongoing work to design schemes within the FRI framework for applications in quantum chemistry. Specifically, we introduce a family of methods for the full configuration interaction (FCI) problem. In deterministic FCI, one first expands the ground state eigenfunction of the many body Schrödinger operator in a basis of Slater determinants (which obey a required antisymmetry property). This results in a finite (but very large) dimensional eigenproblem. For systems with more than a few electrons, this finite dimensional eigenproblem quickly becomes far too large for classical approaches. Our FCI-FRI schemes stochastically impose sparsity during iterations of the power method and can be viewed as a generalization of existing, more direct, applications of DMC to the FCI problem. Performing ground-state calculations on five small molecules at fixed cost, we find that FCI-FRI is up to 8000 times more statistically efficient than the comparable existing approach.

References


Distance Problems for Dissipative Hamiltonian Systems

Christian Mehl, Volker Mehrmann, Michał Wojtylak

Faculty of Mathematics and Computer Science, Jageillonian University, Krakow, Poland
https://apacz.matinf.uj.edu.pl/users/253-michal-wojtylak

Abstract
The talk is based on the papers [1] and [2]. We will be interested in the linear algebra properties of matrix pencils that are associated with linear time-invariant dissipative Hamiltonian descriptor systems of the form

\[ E \dot{x} = (J - R) Q x. \] (1)

where \( J, R \in \mathbb{R}^{n,n}, \ E, Q \in \mathbb{R}^{n,m}, \ m \leq n, \ E^\top Q \geq 0, \ J^\top = -J \) and \( R^\top = R \geq 0, \) (2)

where \( W \geq 0 \) stands for \( W \) being positive semi-definite. Such systems arise in energy based modeling of dynamical systems, see e.g., [3, 4, 5]. Special attention will be put to the case \( m = n, \) \( Q = I_n, \) E.g. a simple RLC network can be modeled by a dissipative Hamiltonian descriptor system of the form

\[
\begin{bmatrix}
G_c C G_c^\top & 0 & 0 \\
0 & L & 0 \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\dot{V} \\
\dot{I}_l \\
\dot{I}_v
\end{bmatrix}
= \begin{bmatrix}
-G_r R_r^{-1} G_r^\top & -G_l & -G_v \\
G_l^\top & 0 & 0 \\
G_v^\top & 0 & 0
\end{bmatrix}
\begin{bmatrix}
V \\
I_l \\
I_v
\end{bmatrix},
\] (3)

where \( L > 0, \ C > 0, \ R_r > 0 \) are real symmetric matrices describing inductances, capacitances, and resistances, respectively and the matrices \( G \) encode the network topology.

The first main result to discuss is the Kronecker canonical form of the pencil associated with (1), see [1].

Theorem. Let \( E, Q \in \mathbb{R}^{n,m} \) satisfy \( E^\top Q = Q^\top E \geq 0 \) and let all left minimal indices of \( \lambda E - Q \) be equal to zero (if there are any). Furthermore, let \( J, R \in \mathbb{R}^{m,m} \) be such that we have \( J = -J^\top, \) \( R \geq 0. \) Then the following statements hold for the pencil \( P(\lambda) = \lambda E - (J - R)Q. \)

(i) If \( \lambda_0 \in \mathbb{C} \) is an eigenvalue of \( P(\lambda) \) then \( \text{Re}(\lambda_0) \leq 0. \)

(ii) If \( \omega \in \mathbb{R} \setminus \{0\} \) and \( \lambda_0 = i\omega \) is an eigenvalue of \( P(\lambda), \) then \( \lambda_0 \) is semi-simple. Moreover, if the columns of \( V \in \mathbb{C}^{m,k} \) form a basis of a regular deflating subspace of \( P(\lambda) \) associated with \( \lambda_0, \) then \( RQV = 0. \) If, additionally, \( Q \) is nonsingular than the statement holds for \( \lambda_0 = 0 \) as well.

(iii) The index of \( P(\lambda) \) is at most two.

(iv) All right minimal indices of \( P(\lambda) \) are at most one (if there are any).

(v) If in addition \( \lambda E - Q \) is regular, then all left minimal indices of \( P(\lambda) \) are zero (if there are any).
The second problem that will be discussed in the talk is the structured distance to singularity, see [2]. Recall that for a regular pencil $\lambda E - A$ we define its (unstructured) distance to singularity as

$$d(\lambda E - A) := \inf\{\|\Delta A, \Delta E\|_F : \lambda (E + \Delta E) - (A + \Delta A) \text{ is singular}\}.$$  

(4)

The problem of computing or estimating this quantity was posed in [6]. It was also showed therein that the basic numerical methods fail. A hope to relax the problem appeared together with the study of structured pencils in [7], however still not simplifying the problem substantially.

It appears that the assumptions in (2) (with $Q = I_n$) are in a sense optimal, i.e. on one hand they cover a large class of modelling problems and allow stable numerical methods, on the other hand computing the distance to singularity is feasible in this class. Namely, if we modify the definition in (4) by adding an additional constraint that the perturbed pencil $\lambda (E + \Delta E) - (A + \Delta A)$ also satisfies the structured properties of (2) (with $Q = I_n$), then this new distance $\tilde{d}(\lambda E - (J - R))$ has effective bounds

$$\lambda_{\min}^{1/2}(-J^2 + R^2 + E^2) \leq \tilde{d}(\lambda E - (J - R)) \leq 2 \cdot \lambda_{\min}^{1/2}(-J^2 + R^2 + E^2),$$

and the minimal singularising perturbation can be constructed explicitly.

In a similar way the structured distance to the nearest high index problem is treated. The results are generalised onto special structured matrix polynomials.

References


Accelerating Parallel Hierarchical Matrix-Vector Products via Data-Driven Sampling

Yuanzhe Xi, Difeng Cai, Lucas Erlandson and Edmond Chow

Department of Mathematics, Emory University
www.math.emory.edu/~yxi26/

Abstract

Hierarchical matrices are scalable matrix representations especially when the matrix entries are defined by a kernel function evaluated between pairs of points. In this talk, we present a new scheme to alleviate the computational bottlenecks present in many hierarchical matrix methods. For general kernel functions, a popular approach to construct hierarchical matrices is through interpolation, due to its efficiency compared to computationally expensive algebraic techniques. However, interpolation-based methods usually lead to larger rank, and do not scale well in higher dimensions. We propose a new data-driven method to resolve these issues. The new method is able to accomplish the rank reduction by using a surrogate for the global distribution of points which is constructed using a hierarchical data-driven sampling [2]. As a result of the lower rank, the construction cost, memory requirements, and matrix-vector product costs decrease. Using state-of-the-art dimension independent sampling [1], the new method makes it possible to tackle problems in higher dimensions. We also discuss an on-the-fly variation of hierarchical matrix construction and matrix-vector products that is able to reduce memory usage by an order of magnitude. This is accomplished by postponing the generation of certain intermediate matrices until they are used, generating them just in time. We provide results demonstrating the effectiveness of our improvements, both individually and in conjunction with each other. For a problem involving 320,000 points in 3D, our data-driven approach reduces the memory usage from 58.6 GB using state-of-the-art methods (762.9 GB if stored dense) to 17.78 GB. In combination with our on-the-fly approach, we are able to reduce the total memory usage to 547.2 MB.

References


Given a non-compact kernel function $K(x, y) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ and two sets of points $X_0, Y_0 \subset \mathbb{R}^d$, a dense kernel matrix is defined as $K(X_0, Y_0)$ made up by $K(x_i, y_j)$ with all $(x_i, y_j) \in X_0 \times Y_0$. Fast algorithms for arithmetics of dense kernel matrices, e.g., rank-structured matrix techniques, are usually based on the low-rank approximation of specific matrix blocks. A general problem for setting up these fast algorithms is the low-rank approximation of a kernel block $K(X_0, Y_0)$, i.e.,

$$K(X_0, Y_0) \approx UV^T \quad \text{with} \quad U \in \mathbb{R}^{\|X_0\| \times r_0}, \ V \in \mathbb{R}^{\|Y_0\| \times r_0}, \ \text{and a small rank } r_0,$$

with $X_0 \times Y_0$ in a pair of compact domains $\mathcal{X} \times \mathcal{Y}$ where $K(x, y)$ can be well approximated by a low-degree degenerate expansion, i.e.,

$$K(x, y) \approx \sum_{i=1}^{r} \psi_i(x) \varphi_i(y), \quad (x, y) \in \mathcal{X} \times \mathcal{Y}. \quad (2)$$

To compute a low-rank approximation (1), analytic methods first construct a degenerate approximation (2) of $K(x, y)$ in $\mathcal{X} \times \mathcal{Y}$ and then plug $X_0$ and $Y_0$ into (2). Meanwhile, algebraic methods only work on the matrix $K(X_0, Y_0)$ itself to construct the approximation.

We present a hybrid analytic-algebraic method that first constructs a low-rank approximation of the quasimatrix $K(X_0, y)$ with $y \in \mathcal{Y}$, i.e., each row of this quasimatrix is a function $K(x_i, y)$ defined in $\mathcal{Y}$, in an interpolative decomposition (ID) form as,

$$K(X_0, y) = \begin{bmatrix} K(x_1, y) \\ K(x_2, y) \\ \vdots \\ K(x_m, y) \end{bmatrix} \approx \begin{bmatrix} u_1^T \\ u_2^T \\ \vdots \\ u_m^T \end{bmatrix} \begin{bmatrix} K(x_{i_1}, y) \\ K(x_{i_2}, y) \\ \vdots \\ K(x_{i_q}, y) \end{bmatrix} = UK(X_{id}, y), \quad y \in \mathcal{Y}, \quad (3)$$

where $X_{id}$ is a subset of $X_0$, $u_i^T$ is of dimension $|X_{id}|$, and $K(x_i, y)$ is approximated by $u_i^T K(X_{id}, y)$. Plugging $Y_0 \subset \mathcal{Y}$ into (3) then gives a low-rank ID approximation of $K(X_0, Y_0)$ as

$$K(X_0, Y_0) \approx UK(X_{id}, Y_0). \quad (4)$$

In general, analytic methods are computationally more efficient but give larger approximation ranks than algebraic methods. The above hybrid analytic-algebraic method lies between the two method classes, giving better approximation ranks than analytic methods (by involving $X_0$ in (3)) while being more efficient than algebraic methods (to be illustrated next).

A quasimatrix ID approximation (3) can be directly computed based on the pivoted QR decomposition [1] of $K(X_p, y)$ which contains inner products between the row functions and can be expensive. To efficiently compute such an ID approximation, we propose the proxy point method [2]. The main idea is to first select a set of so-called proxy points $Y_p$ in $\mathcal{Y}$ and then compute the ID components $U$ and $X_p$ for (3) from an ID approximation of $K(X_0, Y_p)$,

$$K(X_0, Y_p) \approx UK(X_{id}, Y_p), \quad X_{id} \subset X_0, \quad (5)$$
which is algebraically computed by the strong rank-revealing QR (SRRQR) decomposition [3]. For each \(x_i \in X_0\), the \(i\)th row in (5) is written as \(K(x_i, Y_p) \approx u_i^T K(X_{id}, Y_p)\) with its error bounded by a given error threshold. As a result, the defined function approximation in (3),

\[ K(x_i, y) \approx u_i^T K(x_i, y), \quad x_i \in X_0, y \in Y, \]

is guaranteed to be accurate at \(Y_p\). To make this approximation accurate in \(Y\) as well, the key of the proxy point method is a proper selection of proxy points \(Y_p\) that can make sure that when \(|K(x_i, y) - u_i^T K(X_{id}, y)|\) with any \(u_i^T\) and \(X_{id}\) is small at \(Y_p\), it is also small in the whole domain \(Y\). With such a proper point set \(Y_p\), the ID approximation (3) of the quasimatrix \(K(X_p, y)\) computed by the proxy point method can be accurate (up to certain error threshold). Thus, the final defined ID approximation (4) of \(K(X_0, Y_0)\) can be accurate as well.

As can be observed, as long as \(Y_p\) has fewer points than \(Y_0\), the proxy point method can be more efficient than the direct ID approximation of \(K(X_0, Y_0)\) using SRRQR. Based on the approximate degenerateness (2) of \(K(x, y)\) in \(X \times Y\), we can prove that it is possible and also sufficient to only select a small number of proxy points as \(Y_p\), justifying the efficiency of the proxy point method.

It is worth noting that several special instances of the proxy point method have been applied in practice for specific kernel functions, domains, and proxy points [4, 5]. However, these applications are still heuristic and limited. We provide a thorough study of the general proxy point method and address several critical questions: how and under what conditions the method works, rigorous error analysis, and how to select a proper set of proxy points given a kernel function and two domains.

References


Krylov subspace methods for the trust-region subproblem and beyond

Lei-Hong Zhang, Ren-Cang Li, and Chungen Shen

School of Mathematics, Shanghai University of Finance and Economics
http://math.shufe.edu.cn/aa/ab/c2019a43691/page.htm

Abstract

The trust-region method (TRM), as one of the most important approaches in optimization, has been widely used in a broad range of applications for minimizing a continuously differentiable function \( F : \mathbb{R}^n \to \mathbb{R} \). Let \( \mathbf{x}_0 \) be an approximation of the minimizer. Methods based on the second-order information of \( F(\mathbf{x}) \) generally attempt to minimize a local quadratic approximation

\[
q(s) = \frac{1}{2} s^\top Hs + s^\top \mathbf{g} + F(\mathbf{x}_0)
\]

of \( F(\mathbf{x}) \), where \( s = \mathbf{x} - \mathbf{x}_0 \), \( \mathbf{g} = \nabla F(\mathbf{x}_0) \) and \( H \) denotes the gradient and the (approximate) Hessian matrix of \( F(\mathbf{x}) \) at \( \mathbf{x}_0 \), respectively. When \( H \succ 0 \), the (global) minimizer \( s_{\text{opt}} = -H^{-1}\mathbf{g} \) of \( q(s) \) is just the (approximate) Newton step at \( \mathbf{x}_0 \) which is then updated to \( \mathbf{x}_1 = \mathbf{x}_0 + s_{\text{opt}} \). But when \( H \) is indefinite, \( q(s) \) does not have a global minimizer; or even when \( H \succ 0 \), \( s_{\text{opt}} \) can be deemed too big a step to take. It is the treatment for an indefinite Hessian matrix \( H \) or positive definite one but \( s_{\text{opt}} \) deemed too big that leads to the idea of trust region.

TRM restricts the size of the step taken to reach the next approximation \( \mathbf{x}_1 \) of the minimizer of \( F \) by solving the following so-called Trust-Region Subproblem (TRS):

\[
\min_{\|s\|_M \leq \Delta_0} \frac{1}{2} s^\top Hs + s^\top \mathbf{g},
\]

where \( M = M^\top \succ 0 \) is a proper weighting matrix, and \( \Delta_0 > 0 \) is the trust-region radius that is adaptively updated during the iterative process for achieving global convergence. Besides playing a critical role in the unconstrained minimization, TRS (2) by itself also arises in other real-world applications including the Tikhonov regularization for ill-posed problems, graph partitioning problems as well as the Levenberg-Marquardt approach. In these applications, the matrices associated with the quadratic term in TRS often possess special structures and can be very large in size. Moreover, in recent machine learning researches, by means of sub-sampling, there is an increasing interest in using TRM and its variants to minimize training losses and handle the large finite-sum structure of learning problems [2]. All these imply that efficient computational methods for large scale TRS are highly desirable.

Plenty of numerical algorithms have been proposed in the literature for TRS. These algorithms can be basically grouped into two categories: factorization-based algorithms for small-to-medium sized dense problems (see, e.g., [1, 9]) and factorization-free algorithms for large scale and sparse problems (see, e.g., [5, 6, 7, 13]). As the most important factorization-based algorithm, the Moré-Sorensen method [9] is a Newton method, inherently for small-to-medium sized problems. For large scale TRS, a straightforward application is not feasible due to high cost and heavy memory burden. This is where Krylov subspace type projection methods can help, such as the one proposed by Gould et al. [6]. Basically, the Krylov subspace method of Gould et al. first projects a large TRS onto a suitably built Krylov subspace resulting in into a much smaller one which can then be solved by, e.g., the Moré-Sorensen method. This Krylov subspace method for TRS is called GLTR.
Krylov subspace methods have been shown to be very efficient for solving large scale linear systems and eigenvalue problems. There has been a wealth of developments, in both theory and implementation (see, e.g., [3, 12]). As we have mentioned in [14, 15] that TRS lies between the linear system and the eigenvalue problem, and thus the Krylov subspace method GLTR originally suggested in [6] can be viewed as a natural extension of the classical Lanczos method for the linear system and eigenvalue problem to TRS. One may expect that the convergence behavior of such a Krylov subspace method would have been fully understood long time ago, but on the contrary it was remarked in [2] that

“until recently the literature contains no guarantees on the rate of convergence as the dimension of the Krylov subspace dimension grows. This is in contrast to the two predominant Krylov subspace method use-cases, convex quadratic optimization [4, 10, 11] and eigenvector finding [8], where such rates of convergence have been known for decades.”

The purpose of this talk is to present our recent developments [14, 15] concerning convergence behaviors and restarting techniques of Krylov subspace methods for TRS, as well as possible extensions and applications. In particular, we will (i) establish the worst-case linear convergence of GLTR, and then propose new stopping criteria that can be integrated into GLTR for better numerical performance of the trust-region solver TRU in the Fortran routine GLTR in the library GALAHAD\(^1\), and (ii) develop a double-Krylov-subspace restarting procedure to resolve the numerical troubles in GLTR for ill-conditioned TRS.

References


\(^1\)http://www.galahad.rl.ac.uk


Linear algebraic approach to inverse scattering problems

Liliana Borcea, Vladimir Druskin, Mikhail Zaslavsky, Alexander Mamonov and Jörn Zimmerling

Department of Mathematics, University of Michigan
https://sites.google.com/view/jzimmerling/home

Abstract

In an inverse scattering problem one tries to reconstruct spatially varying PDE-coefficients (for instance of a wave-equation) from (sparse) observations of the PDE variable. Inverse scattering problems are at the heart of many important applications like medical imaging, nondestructive testing or geophysical imaging. Inverse scattering problems are inherently non-linear; nevertheless, many solution methods rely on linearization of the problem. Even though the original PDE operator may have an affine relation in the unknown coefficient the PDE variable and thus our observations depend non linearly on this coefficient.

In this contribution, we present a purely linear algebraic approach to inverse scattering problems that allows us to mitigate this non-linearity. The key idea of our approach is to construct data-driven reduced-order models (ROMs) via time-domain Löwner inner products. This ROM interpolates the observations of the PDE variable, which we call the data. Using a series of linear algebraic transforms guarantees the stability of the ROM and allows an interpretation of the ROM as a coarse finite-difference discretization of the underlying PDE operator. As such, it (approximately) inherits the straightforward dependence on the PDE.

We now solve the inverse problem by minimizing the Frobenius distance between the observed ROM and a model ROM, rather than minimizing the mismatch between observed and modeled data, as most conventional methods do. This reformulation leads to fast convergence since the ROM approach mitigates the non linearity of the inverse scattering problem. To be more specific, consider the following hyperbolic system of PDEs

\[
\frac{\partial^2}{\partial t^2}u(t, x) + L(q)L(q)^T u(t,x) = 0, \quad t > 0, \\
u(0, x) = b(x), \quad \partial_t u(0, x) = 0,
\]

(1)

describing the propagation of the wave \(u(t, x)\). Both \(L(q)\) and \(L(q)^T\) are first order partial differential operators in the variable \(x\), with affine dependence on the unknown coefficient \(q(x)\), called the reflectivity. The initial condition \(b(x)\) acts as a source for waves, and for the sake of brevity also as receiver defining time-discrete data as \(D_j = \int_{\Omega} dx b(x)^T u(j\tau, x), j = 0, \ldots, 2n - 1\) at equidistant time steps \(\tau\). The PDE acts as a map \(q \mapsto \{D_j, j = 0, \ldots, 2n - 1\}\) and the inverse scattering problem is to invert this mapping.

Using this data, it is possible to obtain a reduced-order model, difference equation in the discrete time-domain

\[
\frac{u^{\text{ROM}}(j\tau + \tau) - 2u^{\text{ROM}}(j\tau) + u^{\text{ROM}}(j\tau - \tau)}{\tau^2} = L^{\text{ROM}}(q)L^{\text{ROM}}(q)^T u_j^{\text{ROM}}, \quad j \geq 0,
\]

(2)

\[
u^{\text{ROM}}(0) = b \quad u(\tau) = u(-\tau)
\]

where \(L^{\text{ROM}}\) is a matrix and can be obtained from the data using time-domain Löwner inner products and a series of linear algebraic transforms. This reformulation from functions and differential operators to vector and matrices can formally be described by a Galerkin projection of the system.
in equation (1) onto the space spanned by \(\{u(0, x), u(\tau, x), \ldots, u([n-1]\tau, x)\}\). However, we will show that it can be determined from the data interpolation requirement

\[
D_j = \int_\Omega dx \, b(x)^T u(j\tau, x) = b^T u^{\text{ROM}}(j\tau), j = 0, \ldots, 2n - 1, \tag{3}
\]

and the causality requirement that the matrix \([u^{\text{ROM}}(0), \ldots, u^{\text{ROM}}([n-1]\tau)]\) is upper triangular. In this contribution we show finding a solution \(q^*\) to the inverse problem by solving

\[
q^*(x) = \arg \min_{q \in Q} \|L^{\text{ROM}}(q) - L^{\text{ROM}}(q^*)\|_F^2,
\]

is better conditioned and that the objective function is close to linear in \(q\). Therefore, a (nonlinear) Gauss-Newton algorithm converges in few steps compared to minimizing the data missmatch

\[
q^*(x) = \arg \min_{q \in Q} \|D(q) - D(q^*)\|_F^2. \tag{4}
\]

The obtainable resolution in this inverse scattering problem can be related to the projection of a delta function onto the space \(\{u(0, x), u(\tau, x), \ldots, u([n-1]\tau, x)\}\). This resolution determines the search space \(Q\) in the optimization problem.

Most methods to solve inverse scattering problem rely on solving a minimization problem of the form (4) which requires strong regularization to converge to a meaningful result, due to the ill-posed nature of inverse scattering problems. In our proposed methods, the regularization happens on the level of the ROM via a series of linear algebraic truncations and transforms that preserve the structure of the underlying data.

In summary, we present a purely linear algebraic imaging method for a general class of inverse-scattering problems. We make use of the close connections between linear algebraic factorizations and their continuous counterparts as well as the interpretation of finite-difference discretizations as differential operators in the limit of fine discretization.
Computing the restricted singular value decomposition with improved robustness

Ian Zwaan

Applied Computer Science Group, University of Wuppertal
ianzwaan.com

Abstract

The restricted singular value decomposition (RSVD) is a generalization of the ordinary singular value decomposition (SVD or OSVD) to matrix triplets. Applications of the RSVD include, for example, rank minimization of structured perturbations, unitarily invariant norm minimization with rank constraints, low rank approximation of partitioned matrices, restricted total least squares, generalized Gauss–Markov models, etc. See, e.g., Zha [4, 5], De Moor and Golub [3], and their references for more information. The problem is that computing the RSVD accurately and robustly is challenging, and avoiding numerical pitfalls is hard. The goal of the ideas and algorithms presented in this text is to improve upon existing computation methods in these areas, even though some numerical challenges remain.

Unfortunately, computing the RSVD in a numerically sound way is not straightforward. For example, Zha calls his constructive proof in [4, Thm. 3.2] unsuitable for computation, because it uses transformations with potentially ill-conditioned matrices in intermediate steps. Zha addresses this issue by deriving a Kogbetliantz-type algorithm [5], but this algorithm lacks a backward stable method for computing 2-by-2 RSVDs (cf. the 2-by-2 QSVD from Bai and Demmel [1]). Furthermore, the preprocessing phase of his Kogbetliantz-type algorithm requires a sequence of up to four rank decisions, where each depends on the previous one. These dependencies, and the fact that rank determination is an ill-posed problem in floating-point arithmetic, make the preprocessing prone to errors. For example, it would be straightforward to construct a matrix triplet where we should have a clear gap in the singular values for each rank decision in exact arithmetic, but no longer have any gap (or a gap in the right place) for the fourth, or even third, rank decision in floating-point arithmetic. These faulty rank decisions can even show up if we use OSVD instead of, e.g., QR with pivoting, for the rank decisions. Another algorithm is due to Chu, De Lathauwer, and De Moor [2], who present a QR based method which does not require a backward stable 2-by-2 RSVD. However, their method requires a sequence of up to five dependent rank decisions, and may also use nonorthonormal transformations in the preprocessing phase.

I will present a new algorithm for computing the RSVD that consists of four main innovations. The first innovation is a new preprocessing phase that uses at most three rank decisions, and with fewer dependencies. The second innovation is a backward stable 2-by-2 RSVD algorithm like Bai and Demmel’s backward stable 2-by-2 QSVD algorithm [1], which is a crucial part of the Kogbetliantz-type iteration. The third innovation consists of an alternative scaling for the restricted singular triplets that leads to elegant formulas for their computation. The fourth innovation is a useful generalized Schur form of the RSVD, that is obtained at the end of the Kogbetliantz-type iteration and does not require further processing. In particular, this Schur form may suffice for the application at hand, which would allow us to skip the postprocessing phase of the algorithm.
References


