17th GAMM Workshop

Applied and Numerical Linear Algebra

with special emphasis on

High Performance Computing (HPC)

September 07 - 08, 2017
Mathematical Institute
University of Cologne
Cologne, Germany

Book of Abstracts

Supported by

Thursday

	Room 203	Room 005
08:00 - 08:55	**Registration**	
08:55 - 09:00	**Opening**	
09:00 - 09:45	**Wim Vanroose:** Asynchronous Krylov methods with deep pipelines	
09:55 - 10:20	**Siegfried Cools:** On parallel performance and numerical stability of communication hiding pipelined Krylov methods for solving large scale linear systems	**Anastasia Kruchinina:** Parallel recursive polynomial expansion for sparse density matrix construction in electronic structure calculations
10:20 - 10:45	**Sofia Bikopoulou:** Checksum-based fault tolerance for solving large linear systems on multicore architectures	**Parikshit Upadhyaya:** On the convergence factor of the self-consistent field iteration
10:45 - 11:10	**Martin Lanser:** On the use of AMG methods designed for elasticity problems in inexact BDDC methods	**Jascha Knepper:** An adaptive coarse space for the GDSW algorithm
11:10 - 11:40	**Coffee Break**	
11:40 - 12:05	**Martin Köhler:** Efficient computation of the QR decomposition on GPU accelerated architectures	**Emil Ringh:** Krylov methods for low-rank commuting generalized Sylvester equations
12:05 - 12:30	**Jens Saak:** Towards high performance IRKA on hybrid CPU-GPU systems	**Ion Victor Gosea:** Modeling and reduction of hybrid systems
12:30 - 12:55	**Markus Wittmann:** Performance analysis of sparse triangular solve on current hardware architectures	**Martin Kühn:** Adaptive FETI-DP and BDDC methods with a generalized transformation of basis
12:55 - 14:00	**Lunch Break**	
14:00 - 14:45	**Pierre Gosselet:** Error estimation for FETI(DP) and BDD(C)	
14:45 - 15:10	**Christie Louis Alappat:** RACE: Recursive Algebraic Coloring Engine	
15:10 - 15:35	**Ian Zwaan:** Krylov-Schur type restarts for the two-sided Arnoldi method	
15:35 - 16:05	**Coffee Break**	
16:05 - 16:30	**Giampaolo Mele:** Disguised and new Quasi-Newton methods for nonlinear eigenvalue problems	
16:30 - 16:55	**Elias Jarlebring:** The infinite Bi-Lanczos method for nonlinear eigenvalue problems	
17:00 - 18:00	**Meeting of the GAMM Activity Group "Applied and Numerical Linear Algebra"**	
19:00	**Conference Dinner at the Brauhaus Päffgen (Friesenstraße 64–66)**	

Friday

	Room 203	Room 005
09:30 - 10:15	**Oliver Rheinbach:** Nonlinear domain decomposition solvers on the exascale	
10:15 - 10:40	**Jonas Thies:** An optimal domain decomposition method for the C-grid Navier-Stokes Jacobian	
10:40 - 11:05	**Philipp Birken:** Additive W smoothers for multigrid methods for compressible flow problem	
11:05 - 11:35	**Coffee Break**	
11:35 - 12:00	**Robert Luce:** Incremental computation of the block triangular matrix exponential	
12:00 - 12:25	**Behnam Hashemi:** Computing enclosures for the exponential of a symmetric matrix	
12:25 - 12:50	**Andreas Frommer:** A general framework and analysis for block Krylov subspace methods for the computation of matrix functions	
12:50 - 13:00	**Closing Remarks**	

Supported by

RACE: Recursive Algebraic Coloring Engine

Christie Louis Alappat[a] Moritz Kreutzer[a] Jonas Thies[c] Georg Hager[a] Gerhard Wellein[ab]

[a] *Erlangen Regional Computing Center*
[b] *Department of Informatik, Friedrich Alexander Universität, Erlangen-Nürnberg*
[c] *Deutsches Zentrum für Luft- und Raumfahrt, Köln*

1 Abstract

The graph coloring library RACE enables parallelization of kernels carrying distance-k (k≥1) dependencies. Typical examples for such kernels are Gauss-Seidel (GS) scheme with distance-1 dependencies, or sparse matrix transpose vector (SpMTV), symmetric sparse matrix vector (SymmSpMV), and Kaczmarz (KACZ) [1] iteration which have distance-2 dependencies. RACE uses a novel method for graph coloring that achieves high hardware efficiency on modern multi-core architectures. The method is a recursive level-based method that aims at finding optimal permutations while preserving good data locality. A thorough performance analysis demonstrates that our approach outperforms traditional multi-coloring methods in general, Intel MKL [2] implementation of SymmSpMV and even the recursive sparse block (RSB) [3] implementation of SymmSpMV that uses a tailored storage format for such operations. The benefit of RACE for SymmSpMV operation is demonstrated in Fig. 1.

As a full application which benefits from our new method, we present the the FEAST [4] eigenvalue solver in combination with the CGMN [5] ierative linear solver. Replacing the tradition multi-coloring method [6] used in the CGMN solver by our RACE library we achieve a speedup of 2× - 2.5× on modern multi-core processor based shared- memory nodes. This performance boost enables the iterative CGMN solver to be competitive in terms of time to solution with direct methods which are typically used to solve the ill-conditioned linear systems in FEAST when computing large sparse inner eigenvalue problems.

2 Method: A brief overview

The method used in RACE is a recursive level-based approach. Each stage in the recursion has four steps namely level construction, permutation, level coloring and load balancing; as explained below.

2.1 Level Construction : Levels are the basic component of this approach. The $level(n)$ is constructed in a way that only $level(n \pm 1)$ is directly connected. In order to achieve this we execute a BFS (Breadth-First-Search) on the graph of the matrix.

2.2 Permutation : Once the level of each node is marked, the matrix is permuted in ascending order of levels. The permutation is done in a symmetric manner, which allows later for recursive application and updating of the permutation vectors.

2.3 Distance-k Level Coloring : Due to the way in which the levels are created, $level(n)$ and $level(n + k + 1)$ are distance-k independent. This enables to work on these levels in parallel, while working sequentially within a level. This is a major difference compared to existing methods (like CM_RCM [7] , level scheduling [8]), where parallelism is exploited within a level rather than between levels. This kind of parallelization has an advantage of enhanced data locality since each thread works on a level that has strong neighboring connections.

2.4 Load Balancing : The power of the method comes from its load balancing step. This step adds further data locality by using parallelism just to meet hardware requirement and scheduling nearby levels to the same thread. It also takes care to distribute the non-zeros evenly while being compatible with distance-k level coloring (step 2.3). I.e., for a distance-2 dependency at least a distance of two levels has to be preserved between two parallel levels.

The above explained steps are repeated recursively (if necessary) on selected sub-graph of the matrix depending on the parallelism required by the hardware.

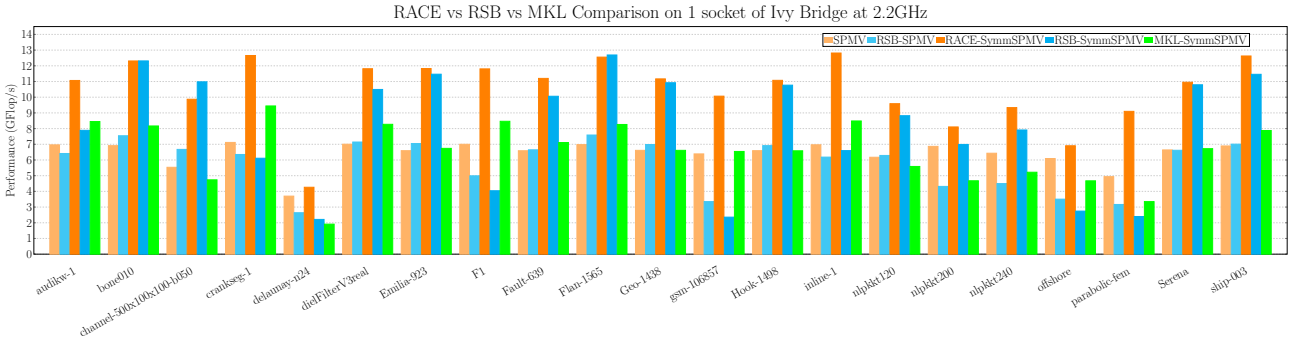

Figure 1: Performance of symmetric sparse matrix vector (SymmSpMV) multiplication for a broad range of test matrices on one socket of a 10-core Ivy Bridge Xeon processor. We compare performance of the general sparse matrix vector (SPMV) multpiplication with implematiations exploiting the symmerty provided by RACE, RSB and Intel MKL.

References

[1] S. Kaczmarz. Angenäherte Auflösung von Systemen linearer Gleichungen. *Bulletin International de l'Académie Polonaise des Sciences et des Lettres*, 35:355–357, 1937.

[2] Intel MKL Library. `https://software.intel.com/en-us/mkl`.

[3] Michele Martone. Efficient multithreaded untransposed, transposed or symmetric sparse matrix-vector multiplication with the recursive sparse blocks format. *Parallel Comput.*, 40(7):251–270, July 2014.

[4] Eric Polizzi. A Density Matrix-based Algorithm for Solving Eigenvalue Problems. *CoRR*, abs/0901.2665, 2009.

[5] Åke Björck and Tommy Elfving. Accelerated projection methods for computing pseudoinverse solutions of systems of linear equations. *BIT Numerical Mathematics*, 19(2):145–163, 1979.

[6] Martin Galgon, Lukas Kraemer, Jonas Thies, Achim Basermann, and Bruno Lang. On the parallel iterative solution of linear systems arising in the FEAST algorithm for computing inner eigenvalues. *Parallel Computing*, 49:153 – 163, 2015.

[7] Kengo Nakajima. Parallel iterative solvers for finite-element methods using an openmp/mpi hybrid programming model on the earth simulator. *Parallel Comput.*, 31(10-12):1048–1065, October 2005.

[8] Jongsoo Park, Mikhail Smelyanskiy, Narayanan Sundaram, and Pradeep Dubey. Sparsifying synchronization for high-performance shared-memory sparse triangular solver. In *Proceedings of the 29th International Conference on Supercomputing - Volume 8488*, ISC 2014, pages 124–140, New York, NY, USA, 2014. Springer-Verlag New York, Inc.

Checksum-Based Fault Tolerance for solving large linear systems on multicore architectures

Sofia Bikopoulou

Institute of Mathematics, Technische Universität Berlin

High-Performance Computing (HPC) systems were initially utilized with executing code on parallel and distributed platforms, providing unique opportunities to industry and academia. The advent of exascale computational architectures has shifted the focus from achieving massive parallelism to using it in a more efficient manner; modern computational models now aim to support accurate and reliable scientific operations.

The challenge of the past decades for HPC applications is to create systems that are both inexpensive and highly reliable. Given the current state-of-the-practice, fewer errors are introduced, but not all errors are prevented. As computational demands in various scientific fields have increased, it is more than essential to provide methods that deliver acceptable level of user-visible service and enable the system's continuous operation, even in the presence of faults. Locating and correcting soft errors and faults due to numerical computations, while minimizing performance loss, is a primary goal that needs to be achieved rapidly.

In this talk we present a technique for tolerating faults when solving large linear systems on HPC platforms with the Generalized Minimum Residual method of Saad and Schultz. The way of achieving fault resilience is by using checkpointing and rollback recovery protocols, in combination with Algorithm-Based Fault Tolerance using Checksums [1, 2]. We provide detection and correction of software faults due to incorrect parallel numerical computations on a large-scale platform.

This is a joint work with Prof. Dr. Volker Mehrmann.

References

[1] G. Bosilca, A. Bouteiller, T. Herault, Y. Robert and J. Dongarra. Composing resilience techniques: ABFT, periodic and incremental checkpointing. International Journal of Networking and Computing, 2015.

[2] D. Tao, S. L. Song, S. Krishnamoorthy, P. Wu, X. Liang, E. Z. Zhang, D. Kerbyson, and Z. Chen. New-sum: A novel online ABFT scheme for general iterative methods. Proceedings of the 25th ACM International Symposium on High-Performance Parallel and Distributed Computing, 2016.

Additive W smoothers for multigrid methods for compressible flow problem

Philipp Birken[a], Jonathan Bull[b] and Antony Jameson[c]

[a] *Lund University, Centre for the Mathematical Sciences, Numerical Analysis*
[b] *Uppsala University, Division of Scientific Computing, Dept. of Information Technology*
[c] *Stanford University, Department of Aeronautics & Astronautics*

We consider multigrid methods for compressible turbulent flow problems. For the Reynolds averaged Navier-Stokes equations (RANS), important progress has been achieved for finite volume discretizations through a new class of preconditioned Runge-Kutta (RK) smoothers [4, 3]. We show that properties of these schemes can be better understood if derived from general additive Runge-Kutta (RK) methods [1]. This gives rise to two classes of preconditioners: Preconditioned additive explicit RK and additive W methods. The latter class can be implemented exactly as the first one, with a suitably chosen preconditioner.

As a preconditioner we look at an SGS preconditioner based on a simplified discretization developed in [4, 2]. A crucial part is a cutoff function for zero eigenvalues, where the cutoff value has to be chosen. We perform a local Fourier analysis of an SGS preconditioner for the Euler equations. The results can be easily understood from the theory of time integration methods and give guidance on how to choose the various parameters of the scheme.

Finally, numerical results for the RANS equations and pitching airfoils are presented that show that with the new schemes, convergence rates below 0.8 can be acchieved.

References

[1] P. Birken, J. Bull, and A. Jameson. A note on terminology in multigrid methods. *PAMM*, 16:721–722, 2016.

[2] A. Jameson. Evaluation of Fully Implicit Runge Kutta Schemes for Unsteady Flow Calculations. *J. Sci. Comput.*, to appear.

[3] S. Langer. Agglomeration multigrid methods with implicit Runge-Kutta smoothers applied to aerodynamic simulations on unstructured grids. *J. Comp. Phys.*, 277:72–100, 2014.

[4] R. C. Swanson, E. Turkel, and C.-C. Rossow. Convergence acceleration of Runge-Kutta schemes for solving the Navier-Stokes equations. *J. Comp. Phys.*, 224(1):365–388, 2007.

On Parallel Performance and Numerical Stability of Communication Hiding Pipelined Krylov Methods for Solving Large Scale Linear Systems

Siegfried Cools[a] and Wim Vanroose[a]

[a] *University of Antwerp, Department of Mathematics & Computer Science, Middelheimlaan 1, 2020 Antwerp, Belgium. siegfried.cools@uantwerp.be, wim.vanroose@uantwerp.be.*

Krylov subspace methods are well-known as efficient solvers for large scale linear systems $Ax = b$. Driven by the transition of hardware towards the exascale regime, research on the scalability of Krylov subspace methods on massively parallel architectures has recently gained increased attention [1]. The main bottleneck for efficient parallel execution of Krylov solvers on large scale hardware is not the sparse matrix-vector product (SPMV) or the AXPY ($y \leftarrow \alpha x + y$) operations, which are (mainly) computed locally, but the communication overhead caused by global reductions in dot-product and norm computations. Each global reduction phase requires the local computation of the dot-product, followed by a reduction tree to gather the scalar result on a single processor, and a subsequent broadcasting of the result to all workers. Time spent by this phase scales as $\log_2(P)$, where P is the number of processors. Over the last decades significant efforts have been made to reduce and/or eliminate the synchronization bottleneck in Krylov subspace methods. In addition to communication avoiding methods [2], pipelined Krylov subspace methods [3, 4, 5] aim to hide global synchronization latency by overlapping the global communication phase(s) by the SPMV(s). Idle core time is thus minimized by performing useful computations simultaneously to the time-consuming global synchronization phase ('communication hiding'). In this talk we give an overview of recent developments in pipelined solvers and we comment on the derivation of the algorithms. Numerical experiments with the pipe-CG [3], pipe-GMRES [4] and pipe-BiCGStab [5] methods are presented, showing that pipelined methods are able to achieve significantly improved parallel scalability compared to standard Krylov methods on present-day HPC hardware. However, the reorganization of the original Krylov subspace algorithm into a pipelined variant introduces additional AXPY operations that are required to recursively compute auxiliary variables. These additional vector operations may influence the numerical stability of the algorithm due to the amplification of local rounding errors in finite precision arithmetic. We investigate the observed loss of maximal attainable accuracy through a numerical analysis of the rounding error propagation behavior [6, 7] and propose various practical countermeasures for the accuracy loss [8, 9].

References

[1] J. Dongarra, M.A. Heroux, and P. Luszczek. HPCG benchmark: a new metric for ranking high performance computing systems. *University of Tennessee, Department EECS, Technical Report UT-EECS-15-736*, 2015.

[2] E. Carson, N. Knight, and J. Demmel. Avoiding communication in nonsymmetric Lanczos-based Krylov subspace methods. *SIAM Journal on Scientific Computing*, 35(5):S42–S61, 2013.

[3] P. Ghysels, T.J. Ashby, K. Meerbergen, and W. Vanroose. Hiding global communication latency in the GMRES algorithm on massively parallel machines. *SIAM Journal on Scientific Computing*, 35(1):C48–C71, 2013.

[4] P. Ghysels and W. Vanroose. Hiding global synchronization latency in the preconditioned Conjugate Gradient algorithm. *Parallel Computing*, 40(7):224–238, 2014.

[5] S. Cools and W. Vanroose. The communication-hiding pipelined BiCGStab method for the parallel solution of large unsymmetric linear systems. *Parallel Computing*, 65:1–20, 2017.

[6] A. Greenbaum and Z. Strakoš. Predicting the behavior of finite precision Lanczos and Conjugate Gradient computations. *SIAM Journal on Matrix Analysis and Applications*, 13(1):121–137, 1992.

[7] Z. Strakoš and P. Tichý. On error estimation in the Conjugate Gradient method and why it works in finite precision computations. *Electronic Transactions on Numerical Analysis*, 13:56–80, 2002.

[8] S. Cools, E.F. Yetkin, E. Agullo, L. Giraud, and W. Vanroose. Analyzing the effect of local rounding error propagation on the maximal attainable accuracy of the pipelined Conjugate Gradients method. *SIAM Journal on Matrix Analysis and Appl., submitted*, preprint: https://arxiv.org/abs/1601.07068, 2017.

[9] S. Cools and W. Vanroose. Numerically stable variants of the communication-hiding pipelined Conjugate Gradients algorithm for the parallel solution of large scale symmetric linear systems. *UAntwerp, Department of Mathematics and Computer Science, Technical Report*, https://arxiv.org/abs/1706.05988, 2017.

A general framework and analysis for block Krylov subspace methods for the computation of matrix functions

Andreas Frommer[a] and Kathryn Lund[a,b] and Daniel B. Szyld[b]

[a] *Bergische Universität Wuppertal, Wuppertal, Germany*
[b] *Temple University, Philadelphia, USA*

We consider the situation where one aims at computing the action of a matrix function $f(A)b_i$ on several vectors $b_i \in \mathbb{R}^n, i = 1, \ldots, s$. The classicial approach would be to approximate from the block Krylov subspace generated by A and the block vector $B = [b_1 \mid \cdots b_s]$, but this introduces a non-neglegible overhead of the order of $\mathcal{O}(ns^2)$ in terms of arithmetic operations per iteration. Alternatives are the so-called global approach which effectively computes $f(I \otimes A)b$ with $b \in \mathbb{R}^{ns}$ obtained by stacking the individual vectors b_i and using the Krylov subspace spanned by $I \otimes A$ and b, and the batched methods which just use a re-arrangement of loops.

In this talk, we show that all these methods can be seen as special cases of a general approach relying on a matrix valued bilinear functional which takes its values in a $*$-algebra with unity. We use this general approach to first derive a convergence theory for the resulting generalized block conjugate gradient methods for solving linear systems and then use this to obtain convergence results for restarted) generalized block Arnoldi methods for computing $f(A)B$ in the case where f is a Stieltjes function.

Modeling and reduction of hybrid systems

Ion Victor Gosea[a] and Athanasios C. Antoulas[b]

[a] *MPI Magdeburg*

[b] *Rice University Houston, MPI Magdeburg, Baylor College of Medicine Houston*

Linear hybrid systems (or LHS's in short) are dynamical systems which are characterized by both continuous and discrete dynamics. The discrete dynamics is determined by a finite-state automaton, while the continuous dynamics is described by sets of linear time varying equations.

The first part is closely related to the finite Moore-automaton which is a finite-state deterministic automaton equipped with inputs and possibly, with outputs. For the second part, the core is represented by the subsystems or discrete modes which are described by a collection of differential or difference equations. The discrete events interacting with the subsystems are governed by a piecewise continuous function, i.e. the switching signal. LHSs can have state-space representations with finitely many state variables, that are used to predict the future behavior of the system.

Hybrid systems offer suitable models for distributed embedded systems design where discrete controls are routinely applied to continuous processes. In some cases, the complexity of such systems might be very high. To cope with this issue, introduce *model order reduction* (MOR) techniques specifically adjusted for such systems.

The proposed reduction method is centered around the definition of new type of infinite Gramians for LHS that resemble the definitions previously encountered for the case of switched, bilinear or stochastic systems. We would like to avoid constructing such Gramians as solutions of systems of linear matrix inequalities (LMIs).

We are interested in the situation for which a common transformation that simultaneously balances all the subsystems of the LHS is either not known, does not exist or it is difficult to attain. We construct a family of transformations, each for a specific mode, based on appropriately defined Gramians. Then, as for the classical linear case, a reduced-order model is computed by truncating the states corresponding to the small diagonal elements of the balanced diagonal Gramians. Finally, several numerical examples illustrate the theoretical formulations.

Error estimation for FETI(DP) and BDD(C)

Pierre Gosselet[a], Valentine Rey[b], Augustin Parret- Freaud[c], and Christian Rey[c]

[a] *LMT, ENS Paris-Saclay/CNRS, France*
[b] *GeM, Univ. Nantes/CNRS, France*
[c] *Safran Tech, France*

We consider linear elasticity problems approximated by the Finite Element (FE) method with the resulting linear systems being solved by non-overlapping domain decomposition methods like FETI(DP) or BDD(C). This presentation deals with the computation of guaranteed upper and lower bounds of the error during the iterations of the solver.

The bounds we consider are based on the error in constitutive equation and on the residual equation. These methods imply to recover certain displacement and stress fields with high regularity on the whole structure. We first show that it is possible to intercept, during the iterations of FETI(DP) or BDD(C), the information necessary for the parallel construction of such fields [3].

Then by a little modification of the classical bounds, we manage to obtain new bounds which separate the algebraic error (due to the use of a DD iterative solver) from the discretization error (due to the FE) [2]. These bounds provides an unbiased criterion to stop the iterations when the solver error is less than the discretization error. They can also be used for the estimation of the error on linear quantities of interest [1].

Assessments on 2D static linear mechanic problems illustrate the relevance of the separation of sources of error and the independence of the bounds with respect to the substructuring. Finally, we steer the iterative solver by an objective of precision on a quantity of interest. The strategy consists in a sequence of solving, adaptive local remeshing and recycling of search directions, in order to reach the desired quality for the quantity of interest with minimal computations.

References

[1] V. Rey, P. Gosselet and C. Rey. *Strict bounding of quantities of interest in computations based on domain decomposition.* CMAME, 287(1):212-228, 2015, doi:10.1016/j.cma.2015.01.009.

[2] V. Rey, P. Gosselet, and C. Rey. *Strict lower bounds with separation of sources of error in non-overlapping domain decomposition methods.* IJNME, 108(9):1007-1029, 2016, doi:10.1002/nme.5244.

[3] A. Parret-Fréaud, V. Rey, P. Gosselet and C. Rey. *Improved recovery of admissible stress in domain decomposition methods - application to heterogeneous structures and new error bounds for FETI-DP.* IJNME, 2017, doi:10.1002/nme.5462.

Computing enclosures for the exponential of a symmetric matrix

Andreas Frommer[a] and **Behnam Hashemi**[b]

[a] *Fakultät für Mathematik und Naturwissenschaften, Bergische Universität Wuppertal, 42097 Wuppertal, Germany*

[b] *Department of Mathematics, Shiraz University of Technology, Modarres BLVD., 71555-313 Shiraz, Iran*

Given a symmetric matrix $A \in \mathbb{R}^{n\times n}$, our goal is to compute an interval matrix that contains all entries of the exact exponential of A [3]. We compare interval arithmetic version of the following methods for computing enclosures to $\exp(A)$:

- spectral decomposition
- contour integration
- Horner evaluation of Taylor series (with scaling and squaring)
- Clenshaw recurrence applied to Chebyshev series (with scaling and squaring).

The first method is based on enclosures for the spectral decomposition of A. Assume that

$$A = VDV^T \in \boldsymbol{V}\boldsymbol{D}\boldsymbol{V}^T, \text{ with } V \in \boldsymbol{V} \in \mathbb{IR}^{n\times n}, D = \mathrm{Diag}(\lambda_1, \ldots, \lambda_n) \in \boldsymbol{D} \in \mathbb{IR}^{n\times n},$$

$VV^T = I$, and $\boldsymbol{V}$ and $\boldsymbol{D}$ are interval matrices which contain the matrices of exact eigenvectors V and eigenvalues D of A, respectively. The exponential of A is then given as

$$\exp(A) = V\exp(D)V^T \in \boldsymbol{V}\exp(\boldsymbol{D})\ \boldsymbol{V}^T,$$

where $\exp(\boldsymbol{D}) = \mathrm{Diag}(\exp(\lambda_{\mathbf{1}}), \ldots, \exp(\lambda_{\mathbf{n}}))$ is a diagonal interval matrix.

The contour integration approach is based on an interval arithmetic version of the N-points trapezoidal rule plus an enclosure bounding the discretization error of trapezoidal rule. Here we rigorously compute sharp error bounds available for periodic integrands that are analytic and bounded on a strip [8].

The third method, due to Goldsztejn and Neumaier [2], is an interval arithmetic version of the scaling and squaring method in combination with the Horner's method.

The fourth method is an interval arithmetic counterpart of the scaling and squaring method combined with the Clenshaw recurrence. Specifically, we use the following four steps:

- Scale the matrix so that its spectrum lies in the canonical interval $[-1, +1]$.
- Enclose the projected Chebyshev series at the scaled matrix with only 15 Chebyshev coefficients [1]. This is done via interval Clenshaw's algorithm adapted to the case that the input is a matrix. Note that the coefficients of Chebyshev expansion of the exponential function are known explicitly [5].
- Rigorously bound the truncation error in chopping Chebyshev series when applied to the scaled matrix. See [7].
- Apply the squaring step via a level 3 BLAS implementation of an enclosure method which is optimal with respect to overestimation suggested in [4].

The difficulty with the first approach above is that computing the interval matrices $\boldsymbol{V}$ and $\boldsymbol{D}$ currently involves $\mathcal{O}(n^4)$ arithmetic operations. See e.g., INTLAB's `verifyeig` routine [6]. In contrast, the last three methods all involve $\mathcal{O}(n^3)$ operations. We compare the above four methods, report numerical examples and highlight challenges ahead.

References

[1] T. A. Driscoll, N. Hale, and L. N. Trefethen, editors, *Chebfun Guide*, Pafnuty Publications, Oxford, 2014.

[2] A. Goldsztejn and A. Neumaier, On the exponentiation of interval matrices, *Reliable Computing* 20 (2014) 53-72.

[3] N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, 2008.

[4] O. Kosheleva, V. Kreinovich, G. Mayer, and H. T. Nguyen, Computing the cube of an interval matrix is NP-hard, Proceedings of the 2005 ACM symposium on Applied computing, ACM, 2005.

[5] J. C. Mason and D. C. Handscomb, *Chebyshev Polynomials*, CRC Press, 2003.

[6] S. M. Rump, INTLAB–interval laboratory, pp. 77-104 in *Developments in Reliable Computing*, Springer, 1999.

[7] L. N. Trefethen, *Approximation Theory and Approximation Practice*, SIAM, 2013.

[8] L. N. Trefethen and J. A. C. Weideman, The exponentially convergent trapezoidal rule, *SIAM Review* 56 (2014) 385-458.

The infinite Bi-Lanczos method for nonlinear eigenvalue problems

Sarah W. Gaaf[a] and **Elias Jarlebring**[b]

[a] *TU Eindhoven, Netherlands*
[b] *KTH Royal Institute of Technology, Sweden*

We consider the problem of determining eigentriples of the nonlinear eigenvalue problem, i.e., the problem to determine non-trivial (x, y, λ) such that

$$T(\lambda)x = 0 \tag{1a}$$
$$T(\lambda)^* y = 0 \tag{1b}$$

where $T : \mathbf{C} \to \mathbf{C}^{n\times n}$ is an analytic function of λ. In this work we propose a *two-sided Lanczos method* for the nonlinear eigenvalue problem. Similar to the linear case, this two-sided approach provides approximations to both the right and left eigenvectors of the eigenvalues of interest. Our method implicitly works with matrices and vectors with infinite size, but because particular (starting) vectors are used, all computations can be carried out efficiently with finite matrices and vectors. We specifically introduce a new way to represent infinite vectors that span the subspace corresponding to the conjugate transpose operation for approximating the left eigenvectors. The approach can be viewed as the Lanczos-adaption of the infinite Arnoldi method [3]. We show that also in this infinite-dimensional interpretation the short recurrences inherent to the Lanczos procedure offer an efficient algorithm regarding both the computational cost and the storage.

The general idea of the approach can be derived from an infinite-dimensional companion linearization. We define the operator $\mathbf{A}$ by the infinite matrix

$$\mathbf{A} := \begin{bmatrix} -\frac{1}{1}T(0)^{-1}T'(0) & -\frac{1}{2}T(0)^{-1}T''(0) & -\frac{1}{3}T(0)^{-1}T^{(3)}(0) & \cdots \\ \frac{1}{1}I & & & \\ & \frac{1}{2}I & & \\ & & \frac{1}{3}I & \\ & & & \ddots \end{bmatrix} \tag{2}$$

and formalize the equivalence with (6). By considering an appropriately defined operator domain. the eigentriplet of $\mathbf{A}$ are equivalent to eigentriplets of the original nonlinear eigenvalue problem, i.e., an eigentriplet of (6) is equivalent to

$$\mathbf{A}\mathbf{v} = \mu\mathbf{v} \tag{3a}$$
$$\mathbf{A}^*\mathbf{w} = \mu\mathbf{w} \tag{3b}$$

where $\lambda = 1/\mu$.

Moreover, the equivalence reveals a structure in the eigevectors. For instance, the left eigenvector $\mathbf{w}$ satisfies

$$\mathbf{w} = \sum_{j=1}^{\infty} (\mathbf{S}^T \otimes I)^{j-1}\mathbf{N}^*\lambda^j z. \tag{4}$$

where $\mathbf{S}$ is a shift-and-scale operator and $\mathbf{N}^*$ an operator containing the blocks $M(0)^{-1}M^{(k)}(0)$.

Our approach is derived by applying the two-sided Lanczos method for linear eigenproblems [1] to the infinite-dimensional operator $\mathbf{A}$. In order to represent the infinite-length vectors, we describe the structure of the vectors needed for the subspaces. The Krylov subspace corresponding to $\mathbf{A}$ and $\mathbf{A}^*$ can be characterized in a way which shows that the elements of the subspace can be represented with a finite number of (finite-length) vectors.

By repeatedly applying this result the inner product of infinite vectors with the particular structure, we can translate every operation in the two-sided Lanczos method, to finite vectors representing the infinite vectors. See the full derivation in [2].

In general, m iterations of our approach requires $\mathcal{O}(nm^3)$ floating-point operations. The impact of this theoretical observation can lessened by certain improvements.

- We reformulate the computationally dominating part of the algorithm in terms of matrix-matrix products, allowing implementation with BLAS level 3 operations, rather than operations with (larger vectors) using BLAS level 2.
- We show how a reformulation of the algorithm can improve the performance if the nonlinear eigenvalue problem can be expressed as a short sum of products of functions and matrices, i.e.,

$$T(\lambda) = T_1 f_1(\lambda) + \cdots + T_k f_k(\lambda),$$

  where $k \ll n$. The value k is small in many applications.

The mathematical software for the simulations are publicly available online.

References

[1] Roland W Freund, Martin H Gutknecht, and Noël M Nachtigal. An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices. *SIAM J. Sci. Comput.*, 14(1):137–158, 1993.

[2] Sarah W. Gaaf, E. Jarlebring The infinite Bi-Lanczos method for nonlinear eigenvalue problems Technical report, 2016, arxiv: http://arxiv.org/pdf/1607.03454, accepted for publication in SIAM J. Sci. Comp.

[3] E. Jarlebring, W. Michiels, and K. Meerbergen. A linear eigenvalue algorithm for the nonlinear eigenvalue problem. *Numer. Math.*, 122(1):169–195, 2012.

[4] V. Mehrmann and H. Voss. Nonlinear eigenvalue problems: A challenge for modern eigenvalue methods. *GAMM-Mitt.*, 27:121–152, 2004.

[5] A. Ruhe. Algorithms for the nonlinear eigenvalue problem. *SIAM J. Numer. Anal.*, 10:674–689, 1973.

[6] H. Voss. Nonlinear eigenvalue problems. In L. Hogben, editor, *Handbook of Linear Algebra, Second Edition*, number 164 in Discrete Mathematics and Its Applications. Chapman and Hall/CRC, 2013.

An Adaptive Coarse Space for the GDSW Algorithm

Alexander Heinlein[a] and Axel Klawonn[a] and **Jascha Knepper**[a] and Oliver Rheinbach[b]

[a] *Mathematisches Institut, Universität zu Köln, Weyertal 86-90, 50931 Köln, Germany,*
{alexander.heinlein, axel.klawonn, jascha.knepper}@uni-koeln.de,
http://www.numerik.uni-koeln.de

[b] *Institut für Numerische Mathematik und Optimierung, Fakultät für Mathematik und Informatik,*
Technische Universität Bergakademie Freiberg, Akademiestr. 6, 09599 Freiberg,
oliver.rheinbach@math.tu-freiberg.de,
http://www.mathe.tu-freiberg.de/nmo/mitarbeiter/oliver-rheinbach

Robust domain decomposition methods for solving second order elliptic problems with large variations in the coefficient rely on the construction of a suitable coarse space. We propose a robust coarse space for the two-level overlapping Schwarz preconditioner, which is an adaptive extension of the energy minimizing coarse space known as GDSW (Generalized Dryja, Smith, Widlund). In particular, we make use of eigenvalue problems using local Schur complements on subdomain edges and faces. The convergence of a corresponding preconditioned Krylov method depends only on a user-specified tolerance and is therefore independent of variations of the coefficient function. The proposed coarse space (AGDSW) reduces to the standard GDSW coarse space if no additional coarse basis functions are used. Furthermore, we suggest several variants of AGDSW to decrease the computational cost.

Efficient Computation of the QR Decomposition on GPU Accelerated Architectures

Martin Köhler[a]

[a] *Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg*

Beside the LU decomposition, the QR decomposition is one of the basic tools in numerical linear algebra, due to its numerical stability. It offers a broad field of applications ranging from, e.g., solving under/over-determined linear systems [3], over preprocessing in generalised eigensolvers [4] to iterative algorithms for spectral division [5].

The most common way of implementation, in public software projects, is the Householder QR decomposition. The performance of this algorithm was improved over the last decades resulting in the availability of a number of BLAS level-3 based implementations. The main idea behind these block algorithms is the grouping of k Householder transformations into the so called compact WY representation. This lifts the rank-one updates used in the classical Householder approach to rank-k updates, which exploit the memory hierarchies present in common computer architectures much more optimally. Similar to the LU decomposition this algorithm can be ported to a hybrid CPU-GPU implementation featuring communication hiding, asynchronous operations, and parallel execution on both CPUs and GPUs.

The hybrid scheme splits the components of the BLAS level-3 algorithm into a CPU and a GPU part. To this end, we assume that A is a block column matrix, i.e.

$$A = \begin{bmatrix} A_1 & A_2 & A_3 & A_4 \end{bmatrix}.$$

The first step of the decomposition computes

$$A_1 = Q_1 R_1$$

on the CPU and, on the GPU, updates the trailing matrix according to

$$Q_1^T \begin{bmatrix} A_2 & A_3 & A_4 \end{bmatrix} = \begin{bmatrix} Q_1^T A_2 & Q_1^T A_3 & Q_1^T A_4 \end{bmatrix} = \begin{bmatrix} \tilde{A}_2 & \tilde{A}_3 & \tilde{A}_4 \end{bmatrix}.$$

As soon as $\tilde{A}_2$ gets ready on the accelerator it can be transferred back to the host, where it can be decomposed into $\tilde{A}_2 = Q_2 R_2$, while the GPUs still works on the updates of $\tilde{A}_3$ and $\tilde{A}_4$. Obviously, the width of the blocks in A should be chosen such that the CPU has decomposed the next block before the update of the trailing matrix columns finishes. In view of the emerging properties of modern hybrid architectures this can present a problem whenever there is a considerably large performance gap between the CPUs and the GPUs, in favour of the GPUs. Then, on the one hand, the width of the blocks needs to be small to reduce the workload on the CPUs but, on the other hand, it has to be large enough for the GPUs to perform the rank-k update efficiently. Furthermore, the relative workload of the CPUs and the GPUs changes during the algorithm, since the trailing submatrix becomes smaller and smaller, which increases this problem even more.

In our contribution, we will compare how this problem influences the QR decomposition of a square matrix on different architectures like an Intel Haswell based system with Nvidia K20 accelerators and an IBM POWER 8 system with P100 accelerators, as well as the new NVlink interconnect. The later is a system the very well demonstrates the effect of the increasing performance gap between CPUs and GPUs. As possible solutions we accelerate the host part of the algorithm by integrating a parallel Tall-Skinny QR (TSQR) decomposition with Householder-Reconstruction [1] and introduce automatic block size adjustment, to cope with the decreasing GPU load due to the shrinking trailing submatrix. Numerical experiments show the advantage of our proposed implementation over current state of the art implementations in standard libraries such as MAGMA (see e.g. [2]).

References

[1] G. Ballard, J. Demmel, L. Grigori, M. Jacquelin, N. Knight, and H. Nguyen, *Reconstructing householder vectors from tall-skinny QR*, J. Parallel Distr. Com., 85 (2015), pp. 3–31.

[2] J. Dongarra, M. Gates, A. Haidar, J. Kurzak, P. Luszczek, S. Tomov, and I. Yamazaki, *Accelerating numerical dense linear algebra calculations with gpus*, Numerical Computations with GPUs, (2014), pp. 1–26.

[3] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, fourth ed., 2013.

[4] C. B. Moler and G. W. Stewart, *An algorithm for generalized matrix eigenvalue problems*, SIAM J. Numer. Anal., 10 (1973), pp. 241–256.

[5] X. Sun and E. S. Quintana-Ortí, *Spectral division methods for block generalized Schur decompositions*, Mathematics of Computation, 73 (2004), pp. 1827–1847.

Parallel recursive polynomial expansion for sparse density matrix construction in electronic structure calculations

Anastasia Kruchinina[a], Emanuel H. Rubensson[a] and Elias Rudberg[a]

[a] *Division of Scientific Computing, Department of Information Technology, Uppsala University, Sweden*

Density matrix recursive polynomial expansion is a powerful tool for linear scaling construction of the density matrix in electronic structure calculations. With this approach, the computational cost scales linearly with system size provided that small matrix elements are removed during the recursive expansion [1, 2]. In general, the obtained sparsity patterns are irregular and are not known beforehand. Therefore load balance in parallel computations is hard to achieve using traditional parallel programming models. We parallelize the recursive expansion using a recently developed parallel block-sparse matrix library [3], based on the *Chunks and Tasks* programming model [4, 5]. The programming model is general and suits various problems in scientific computing that require parallelizing dynamic hierarchical algorithms such as block-sparse matrix operations. The user divides data and work into chunks and tasks which are dynamically distributed by the runtime library to the physical resources. In the block-sparse matrix library matrices are represented by sparse quad-trees of chunks, where chunks at the leaf level contain submatrices represented using a block-sparse structure. We evaluate the performance of our implementation of the recursive polynomial expansion using problems coming from electronic structure calculations, investigating linear scaling with system size and parallel strong and weak scaling.

References

[1] A. Kruchinina, E. Rudberg, E. H. Rubensson, Parameterless stopping criteria for recursive density matrix expansions, J. Chem. Theory Comput., 12 (2016), pp. 5788–5802

[2] E. Rudberg, E. H. Rubensson, and P. Sałek, Kohn–Sham density functional theory electronic structure calculations with linearly scaling computational time and memory usage, J. Chem. Theory Comput., 7 (2011), pp. 340–350.

[3] E.H. Rubensson, E. Rudberg, Locality-aware parallel block-sparse matrix-matrix multiplication using the Chunks and Tasks programming model, Parallel Computing, 57 (2016), pp. 87–106.

[4] E.H. Rubensson, E. Rudberg, Chunks and Tasks: A programming model for parallelization of dynamic algorithms, Parallel Computing, 40 (2014), pp. 328–343.

[5] E.H. Rubensson, E. Rudberg, CHT-MPI: an MPI-based Chunks and Tasks library implementation, version 1.2, `http://www.chunks-and-tasks.org`.

Adaptive FETI-DP and BDDC Methods with a Generalized Transformation of Basis

Martin Kühn[a], Axel Klawonn[a], and Oliver Rheinbach[b]

[a] *Mathematisches Institut, Universität zu Köln, Weyertal 86-90, 50931 Köln, Germany*
[b] *Technische Universität Bergakademie Freiberg, Fakultät für Mathematik und Informatik, Institut für Numerische Mathematik und Optimierung, 09596 Freiberg, Germany*

Adaptive Coarse Spaces for the FETI-DP (Finite Element Tearing and Interconnecting) and BDDC (Balancing Domain Decomposition by Constraints) domain decomposition methods are considered. These adaptive coarse spaces are based on the computation of small local eigenvalue problems. A special emphasis is put on the three dimensional case and on materials with heterogeneous material parameters where coefficient jumps are not aligned with the interface of the domain decomposition. Our new approach is based on solving local eigenvalue problems on faces, enriched by a selected, small number of additional local eigenvalue problems on edges. The additional edge eigenvalue problems make the method provably more robust. The introduction of the additional edge eigenvalue problems yields a condition number that only depends on the tolerance of the local eigenvalue problems and some properties of the domain decomposition and is independent of discontinuities of the material parameters. Until recently, we have implemented the adaptive constraints by a balancing (or deflation) approach. In the present work, we use a transformation of basis approach in combination with partial subassembly. Compared to the case of constant coefficients on subdomains, the transformation of basis approach has to be adapted for general heterogeneous materials. Numerical results are shown for linear elasticity and composite materials supporting our theoretical findings.

References

[1] A. Klawonn, M. Kühn, O. Rheinbach, *Adaptive Coarse Spaces for FETI-DP in Three Dimensions*, SIAM J. Sci. Comput., Vol. 38(5), pp. A2880-A2911 (2016).

[2] A. Klawonn, M. Kühn, O. Rheinbach, *FETI-DP and BDDC Methods with a Generalized Transformation of Basis For Heterogeneous Problems: Connections to Deflation*. Submitted April 2017.

[3] A. Klawonn, M. Kühn, O. Rheinbach,*Adaptive FETI-DP and BDDC methods with a Generalized Transformation of Basis for Heterogeneous Problems*. Submitted July 2017.

On the use of AMG methods designed for elasticity problems in inexact BDDC methods

Martin Lanser[a], Axel Klawonn[a], and Oliver Rheinbach[b]

[a] *Mathematisches Institut, Universität zu Köln, Weyertal 86-90, 50931 Köln, Germany*
[b] *Technische Universität Bergakademie Freiberg, Fakultät für Mathematik und Informatik, Institut für Numerische Mathematik und Optimierung, 09596 Freiberg, Germany*

Traditionally, domain decomposition methods as BDDC (Balancing Domain Decomposition by Constraints) or FETI-DP (Finite Element Tearing and Interconnecting - Dual Primal) use sparse direct solvers as building blocks, i.e., to solve local subdomain problems and/or the coarse problem. Especially the direct solution of the coarse problem can lead to an insufficient weak scaling behavior, since the coarse space grows proportionally with the number of processors and subdomains. Therefore, the sparse direct solvers are often replaced by spectrally equivalent preconditioners without loss of convergence speed. In BDDC domain decomposition methods, such approaches have first been introduced in [1, 2], and have since then successfully been used in large parallel codes.

In this talk, highly scalable implementations of different inexact BDDC (Balancing Domain Decomposition by Constraints) variants are presented, and scalability results for linear elasticity problems in two and three dimensions for up to 131 072 computational cores of the JUQUEEN BG/Q are shown. In this methods, the inverse action of the partially coupled stiffness matrix is replaced by V-cycles of an AMG (algebraic multigrid) method. The use of classical AMG for systems of PDEs, based on a nodal coarsening approach is compared with a recent AMG method using an explicit interpolation of the rigid body motions (global matrix approach; GM) [3]. It is illustrated that for systems of PDEs an appropriate AMG interpolation is mandatory for fast convergence, i.e., using exact interpolation of rigid body modes in elasticity.

References

[1] Jing Li and Olof B. Widlund, *On the use of inexact subdomain solvers for BDDC algorithms*, Comput. Meth. Appl. Mech. Engrg., 196: 1415–1428, 2007.

[2] Clark R. Dohrmann, *An approximate BDDC preconditioner*, Numer. Linear Algebra Appl., 14(2):149–168, 2007.

[3] Allison H. Baker, Axel Klawonn, Tzanio Kolev, Martin Lanser, Oliver Rheinbach, and Ulrike Meier Yang, *Scalability of Classical Algebraic Multigrid for Elasticity to Half a Million Parallel Tasks*, Software for Exascale Computing - SPPEXA 2013-2015, Springer International Publishing, 113–140, 2016.

Incremental computation of the block triangular matrix exponential

Daniel Kressner[a], **Robert Luce**[a] and Francesco Statti[a]

[a] *Ecole polytechnique fédérale de Lausanne*

We study the problem of computing the matrix exponential of a block triangular matrix

$$G_n = \begin{bmatrix} G_{0,0} & G_{0,1} & \cdots & G_{0,n} \\ & G_{1,1} & \cdots & G_{1,n} \\ & & \ddots & \vdots \\ & & & G_{n,n} \end{bmatrix} \in \mathbb{R}^{N \times N},$$

(all diagonal matrices $G_{i,i}$ are diagonal) in a peculiar way: Block column by block column, from left to right. In other words, we wish to compute the sequence of matrix exponentials

$$\exp(G_0),\ \exp(G_1),\ \exp(G_2),\ \ldots, \tag{5}$$

which are the leading portions of $\exp(G_n)$, until some termination criterion is satisfied.

The need for such an evaluation scheme arises naturally in the context of option pricing in polynomial diffusion models. In this setting a discretization process produces a sequence of nested block triangular matrices, and their exponentials are to be computed at each stage, until a dynamically evaluated criterion allows to stop.

Our algorithm is based on *scaling and squaring*. By carefully reusing certain intermediate quantities from one step to the next, we can efficiently compute the sequence (5) of matrix exponentials. In our implementation the computational overhead induced by this peculiar evaluation order is very moderate: Asymptotically the number of operations for computing $\exp(G_n)$ increases only by a logarithmic factor from $\mathcal{O}(\log(\|G_n\|)N^3)$ for the standard scaling and squaring algorithm to $\mathcal{O}(\log(\|G_n\|)^2N^3)$.

References

[1] D. Kressner, R. Luce & F. Statti. *Incremental computation of block triangular matrix exponentials with application to option pricing*, 2017, arXiv:1703.00182

Disguised and new Quasi-Newton methods for nonlinear eigenvalue problems

E. Jarlebring and A. Koskela and **G. Mele**

KTH Royal Institute of Technology, Stockholm, Sweden.

We consider the nonlinear eigenvalue problem (NEP) of the type: given $M : \mathbb{C} \to \mathbb{C}^{n\times n}$ a holomorphic function, determine $(\lambda, v) \in \mathbb{C} \times (\mathbb{C}^n \setminus \{0\})$ such that

$$M(\lambda)v = 0. \tag{6}$$

The vector v is called eigenvector and the scalar λ eigenvalue. Our approach is based on the well-know augmented formulation of the NEP (6), stated as finding the zeros of a nonlinear system of equations:

$$F\left(\begin{bmatrix} v \\ \lambda \end{bmatrix}\right) := \begin{bmatrix} M(\lambda)v \\ c^H v - 1 \end{bmatrix} = 0 \tag{7}$$

which is equivalent to (6) if $c \in \mathbb{C}^n$ is not orthogonal to the eigenvector v. The problem (7) can be approached with Newton's method. One of the most common techniques to improve the convergence or efficiency of Newton's method is to replace the Jacobian matrix with a different matrix. We investigate which types of approximations of the Jacobian matrix lead to competitive algorithms, and provide convergence theory.

Among quasi-Newton methods, the most common variation is to keep the Jacobian matrix constant. The factorization of this matrix can be precomputed before carrying out the iterations. This is beneficial, e.g., in situations where the problem stems from a discretization of a PDE, as the resulting system is often large and the Jacobian matrix is sparse with a structure allowing a sparse LU-factorization. There are various flavors of Newton's method available in the literature for this class of NEPs. Some of these methods do have the property that the matrix in the linear system to be solved in every iteration remains constant. However, these methods are in general not seen as Jacobian matrix modifications of Newton's method applied to (7), but are often derived from quite different viewpoints.

We derive new algorithms and also show that several well-established methods for NEPs can be interpreted as quasi-Newton methods, and thereby provide insight to their convergence behavior. In particular, we establish quasi-Newton interpretations of Neumaier's residual inverse iteration [5] and Ruhe's method of successive linear problems [6]. The convergence analysis is based on theory for quasi-Newton methods and Keldysh's theorem [2] for NEPs [3, 4].

References

[1] E. Jarlebring and A. Koskela and G. Mele. Disguised and new Quasi-Newton methods for nonlinear eigenvalue problems. arXiv preprint arXiv:1702.08492, 2017.

[2] R. Mennicken and M. Möller. Non-self-adjoint boundary eigenvalue problems. Gulf Professional Publishing, 2003.

[3] W.-J. Beyn. An integral method for solving nonlinear eigenvalue problems. Linear Algebra Appl. 2012.

[4] D. B. Szyld and F. Xue. Local convergence of Newton-like methods for degenerate eigenvalues of nonlinear eigenproblems. Numer. Math., 2015.

[5] A. Neumaier. Residual inverse iteration for the nonlinear eigenvalue problem. SIAM J. Numer. Anal., 1985.

[6] A. Ruhe. Algorithms for the nonlinear eigenvalue problem. SIAM J. Numer. Anal., 1973.

Nonlinear Domain Decomposition Solvers on the Exascale

Oliver Rheinbach[a], Axel Klawonn[b], Martin Lanser[b], and Matthias Uran[b]

[a] *Technische Universität Bergakademie Freiberg, Fakultät für Mathematik und Informatik, Institut für Numerische Mathematik und Optimierung, 09596 Freiberg, Germany*

[b] *Mathematisches Institut, Universität zu Köln, Weyertal 86-90, 50931 Köln, Germany*

Parallel nonlinear solvers are at the core of nonlinear, implicit simulations in computational sciences and engineering, e.g., using finite elements in nonlinear mechanics. Nonlinear domain decomposition methods are divide-et-impera strategies which make use of concurrent nonlinear subdomain problems to more efficiently solve the original nonlinear problem. In addition to the nonlinear subdomain problems also a coarse problem is solved, which introduces global coupling of the local nonlinear problems. In the SPPEXA project EXASTEEL such solvers are applied to solve heterogeneous, nonlinear structural mechanics problems. We present parallel scalability of versions of nonlinear domain decomposition methods to millions of MPI ranks.

Krylov methods for low-rank commuting generalized Sylvester equations

Elias Jarlebring[a], Giampaolo Mele[a], Davide Palitta[b], and **Emil Ringh**[a]

[a] *Department of Mathematics, KTH Royal Institute of Technology, Stockholm, Sweden*
[b] *Dipartimento di Matematica, Università di Bologna, Bologna, Italy*

We consider a generalization of the Sylvester equation defined by

$$AX + XB^T + \sum_{i=1}^{m} N_i X M_i^T = C_1 C_2^T, \tag{1}$$

and its special case, the generalized Lyapunov equation

$$AX + XA^T + \sum_{i=1}^{m} N_i X N_i^T = CC^T, \tag{2}$$

where $A, B, N_i, M_i \in \mathbb{R}^{n\times n}$ for $i = 1, \dots, m$, and $C_1, C_2, C \in \mathbb{R}^{n\times r}$. These matrix equations appear naturally in many applications; e.g., for bilinear- and stochastic control systems, where the singular values of the solution to (2) relates to reachability and observability of certain states which is used in model order reduction techniques [1]. Moreover, equations of the type (1) arise in the context of discretizations of certain partial differential equations on rectangular domains, see e.g., [2, 3].

More precisely, we consider a specific class of large-scale problems of the form (1). Let $\mathcal{L}(X) := AX + XB^T$ be the Sylvester operator and let $\Pi(X) := \sum_{i=1}^{m} N_i X M_i^T$. We assume that $\rho(\mathcal{L}^{-1}\Pi) < 1$, where $\rho(\cdot)$ denotes the spectral radius. Furthermore, we assume that the right-hand side of (1) has low rank in the sense that $r \ll n$, and that the matrix coefficients are low-rank commuting, i.e., the commutators $[A, N_i] := AN_i - N_iA$, and $[B, M_i] := BM_i - M_iB$ are matrices of low rank, for $i = 1, \dots, m$.

In this work we present a computational approach for this class of generalized Sylvester equations. One recent successful approach to solve large-scale matrix equations is based on projection methods. A necessary (but not sufficient) condition for the success of projection methods is low-rank approximability, i.e., the solution $X \in \mathbb{R}^{n\times n}$ can be approximated by $\hat{X}$ where $\|X - \hat{X}\|$ is small and $\hat{X}$ has low rank. We show the existence of such low-rank approximations for our class of problems. Projection methods require a choice of subspace, where Krylov subspaces is one of the most common choices (but certainly not the only one). Krylov subspaces require a choice of a starting vector/block. We show how the low-rank commutation property can be exploited to derive natural and efficient choices of staring blocks. The derived properties are combined, leading to a complete structure exploiting algorithm.

We present numerical simulations and compare with other methods for generalized Lyapunov equations [4, 5]. The code for the simulation is provided online for reproducibility, see link in [6].

References

[1] P. Benner, and T. Damm. Lyapunov equations, energy functionals, and model order reduction of bilinear and stochastic systems. SIAM J. Control and Optim. 49 (2) (2011): 686-711.

[2] D. Palitta, and V. Simoncini. Matrix-equation-based strategies for convection-difusion equations. BIT 56.2 (2016): 751-776.

[3] E. Ringh, G. Mele, J. Karlsson, and E. Jarlebring. Sylvester-based preconditioning for the waveguide eigenvalue problem. Accepted for publication in Linear Algebra Appl. (2017).

[4] P. Benner, and T. Breiten. Low rank methods for a class of generalized Lyapunov equations and related issues. Numer. Math. 124 (3) (2013): 441-470.

[5] S. D. Shank, V. Simoncini, and D. B. Szyld. Efficient low-rank solution of generalized Lyapunov equations. Numer. Math. 134 (2) (2016): 327-342.

[6] E. Jarlebring, G. Mele, D. Palitta, and E. Ringh. Krylov methods for low-rank commuting generalized Sylvester equations. Preprint arXiv:1704.02167 (submitted).

Towards high performance IRKA on hybrid CPU-GPU systems

Jens Saak[a] and Georg Pauer[b] and Kapil Ahuja[c] and Ruchir Garg[c]

[a] *Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany*
[b] *Otto von Guericke University, Magdeburg, Germany*
[c] *Indian Institute of Technology Indore, India*

The iterative rational Krylov algorithm (IRKA)[3] is an attractive method for the model order reduction of linear dynamical systems

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{8}$$

due to the fact that it is (locally) optimal in the H_2 system norm and thus also minimizes the error in the system output in the L_∞ norm. Here, we limit ourselves to the case where (8) is stable with matrices $E, A \in \mathbb{R}^{n\times n}$ (E invertible), $B \in \mathbb{R}^{n\times 1}$ and $C \in \mathbb{R}^{1\times n}$ and $D = 0$ to simplify notation. We seek to approximate (8) by a much smaller system

$$\begin{aligned} \hat{E}\dot{\hat{x}}(t) &= \hat{A}\hat{x}(t) + \hat{B}u(t), \\ \hat{y}(t) &= \hat{C}\hat{x}(t) + Du(t), \end{aligned} \tag{9}$$

with $\hat{E} := W_k^T E V_k$, $\hat{A} := W_k^T A V_k \in \mathbb{R}^{k\times k}$, $\hat{B} := W_k^T B \in \mathbb{R}^{k\times 1}$ and $\hat{C} := CV_k \in \mathbb{R}^{1\times k}$ and $k \ll n$.

To this end IRKA computes a sequence of Rational Krylov subspaces

$$\begin{aligned} \mathcal{K}_k(E, A, B; \sigma) &= \operatorname{span}\{(\sigma_1 E - A)^{-1}B, \ldots, (\sigma_k E - A)^{-1}B\} = \operatorname{span} V_k, \\ \mathcal{K}_k(E^T, A^T, C^T; \sigma) &= \operatorname{span}\{(\sigma_1 E^T - A^T)^{-1}C^T, \ldots, (\sigma_k E^T - A^T)^{-1}C^T\} = \operatorname{span} W_k. \end{aligned}$$

Here $\sigma \in \mathbb{C}^k$ is taken as the mirrored system poles of one reduced order model (9) to generate the next one until convergence.

Thus, in each step of the algorithm a sequence of k linear systems of equations needs to be solved. We want to investigate the optimal implementation on current hybrid CPU-GPU systems and present 3 candidate approaches using variants of the BiCG iteration as the solver for these linear systems. The first approach tries to exploit preconditioner realignment following [2], the second exploits a Krylov recycling in the spirit of [1] and the last one modifies the BiCG to run for all linear systems at the same time and use matrix matrix products rather than several matrix vector products.

References

[1] K. Ahuja, E. de Sturler, S. Gugercin, and E. R. Chang. Recycling BiCG with an application to model reduction. *SIAM J. Sci. Comput.*, 34(4):A1925–A1949, 2012.

[2] H. Anzt, E. Chow, J. Saak, and J Dongarra. Updating incomplete factorization preconditioners for model order reduction. *Numer. Algorithms*, 73(3):611–630, February 2016.

[3] S. Gugercin, A. C. Antoulas, and C. A. Beattie. $\mathcal{H}_2$ model reduction for large-scale dynamical systems. *SIAM J. Matrix Anal. Appl.*, 30(2):609–638, 2008.

An Optimal Domain Decomposition Method for the C-Grid Navier-Stokes Jacobian

Jonas Thies[a], Sven Baars[b], Mark v.d. Klok[b] and Fred Wubs[b]

[a] *German Aerospace Center (DLR), Simulation and Software Technology, Linder Höhe, 51147 Köln*
[b] *Institute of Mathematics and Computer Science, University of Groningen, the Netherlands*

The Arakawa C-grid scheme is perhaps the most well-known discretization of the incompressible Navier-Stokes equations. The different variables (velocity components and pressure) are placed on the faces and in the center of the grid cells, respectively, to achieve good conservation and stability properties. In [1] an optimal ordering of the variables for the sequential LU-decomposition of the resulting Jacobian was developed based on observations how fill is generated for such matrices during Gaussian elimination. In [2] the method was used in a domain decomposition approach and extended by a robust dropping strategy that leads to a preconditioner achieving a grid-independent convergence rate of GMRES. Structure-preserving properties of this incomplete LU (ILU) factorization allow recursive application and achieving optimal complexity of $\mathcal{O}(N \log N)$ for scalar problems already.

In this talk we show that this goal can be achieved for the 3D C-grid Navier-Stokes equations as well by using a special choice of space-filling subdomain shapes (parallelepipeda). We demonstrate results with this novel partitioning approach for both direct factorization and the multi-level ILU method on several thousand CPU cores. Possible applications include fully implicit time integration and bifurcation analysis of fluid dynamics problems.

References

[1] A.C. De Niet and F.W. Wubs. Numerically stable LDL^T-factorization of F-type saddle point matrices. IMA J. Num. Anal., 2009.

[2] F.W. Wubs and J. Thies. A robust two-level incomplete factorization for (Navier-)Stokes saddle point matrices. SIAM. J. Matrix Anal. & Appl., 2011.

On the convergence factor of the self-consistent field iteration

Parikshit Upadhyaya[a] and Elias Jarlebring[a]

[a] *Department of Mathematics, KTH Stockholm, Sweden*

The self-consistent field (SCF) iteration is an iterative algorithm used to solve a class of non-linear eigenvalue problems of the form $H(X)X = X\Lambda, X^TX = I$, where $H : \mathbb{R}^{n\times k} \to \mathbb{R}^{n\times n}, X \in \mathbb{R}^{n\times k}$ and $\Lambda \in \mathbb{R}^{k\times k}$ is a diagonal matrix. These problems arise frequently in the context of quantum chemistry and electronic structure calculations as the discretized Kohn-Sham and Hartree-Fock equations, where we are interested in computing the k smallest eigenvalues. Sufficient conditions for convergence of SCF exist in the current literature, e.g. in, [1], [2] and [3]. In contrast to these results, we provide a convergence theory based on the analysis of SCF as a fixed point iteration with the density matrix XX^T as the state of the algorithm. This allows us to provide necessary and sufficient conditions for local convergence, and an exact formula for the convergence factor. We make several interpretations of the convergence factor formula, e.g., in terms of $|\lambda_k - \lambda_{k+1}|$, also known as the HOMO-LUMO gap. The exact characterization of the Jacobian of the fixed-point iteration to allows us to improve convergence, and low-rank approximations appear to give us competitive improvements of this approach. Our numerical experiments with various problem sizes and different values of k confirm our theoretical predictions for the convergence factor and the acceleration.

References

[1] C. Yang, W. Gao, and J. C. Meza. *On the convergence of the self-consistent field iteration for a class of nonlinear eigenvalue problems.* SIAM J. Matrix Anal Appl., 30(4):1773-1788, 2009

[2] X. Liu, X. Wang, Z. Wen and Y. Yuan. *On the convergence of the self-consistent field iteration in Kohn-Sham density functional theory.* SIAM J. Matrix Anal Appl., 35(2):546-558, 2014

[3] E. Cances and C. Le Bris. *On the convergence of SCF algorithms for the Hartree-Fock equations.* M2AN, Math. Model. Numer. Anal., 2000

Asynchronous Krylov methods with deep pipelines

Wim Vanroose[a]

[a] *Department of Mathematics and Computer Science, U. Antwerpen, Belgium*

The performance of Krylov method in the strong scaling limit is suffering due to the synchronization latencies associated with the dot-products. We review the development of pipelined Krylov methods that exploit asynchronous communication to overlap communication and computation. For example, by reordering the operations in the conjugate gradients it is possible to execute the dot-product simultaneously with the sparse matrix vector products. We also introduce deep pipelines where the dot-products are overlapping with multiple sparse matrix-vector products. These deep pipelines solve most of the scaling problems. We also discuss the propagation of rounding errors.

Performance Analysis of Sparse Triangular Solve on Current Hardware Architectures

Markus Wittmann[a] Georg Hager[a] Radim Janalik[b] Axel Klawonn[d] Oliver Rheinbach[e] Olaf Schenk[b] Gerhard Wellein[c]

[a]Erlangen Regional Computing Center, University of Erlangen-Nuremberg, Germany
[b]Institute of Computational Science, Faculty of Informatics, Universitá della Svizzera italiana, Switzerland
[c]Department of Computer Science, University of Erlangen-Nuremberg, Germany
[d]Faculty of Mathematics and Natural Sciences University of Cologne, Germany
[e]Faculty of Mathematics and Computer Science, TU Bergakademie Freiberg, Germany

We analyse and model the performance of the sparse triangular solve step for sparse matrices as it is implemented in PARDISO [1]. We employ the Roofline performance model [2], which assumes code performance is either limited by floating point performance of the CPU or the attainable memory bandwidth. The model is adapted to cover the different serial and parallel phases during solve and gives an initial prediction of achievable performance. We then apply the ECM performance model [3, 4] for a more accurate prediction of the solve step. The ECM model can be seen as an extension to the Roofline model and allows us to analyse the runtime contributions stemming from the execution inside the core and the different cache levels in the memory hierarchy. Hereby e.g., the impact of loop unrolling is studied. From this single core performance analysis we predict the scaling behavior over the processor's cores.

Furthermore we establish an ECM model for a best case implementation. This is guided by the previous analysis of PARDISO's solve step. We include vectorization, which is only partially possible as due to the sparse nature of the factor indirect accesses occur.

Finally the performance of the sparse solve step is evaluated on current Intel architectures like Ivy Bridge, Haswell, Broadwell and Knights Landing as well as the recent AMD Ryzen processor and a SX-ACE node, shown in Fig. 1. Additionally we implemented the solve step with intrinsics according to our best case assumptions. Results are compared against the Roofline and ECM predictions. In general we receive a much more precise prediction from the ECM model. However due to the indirect addressing effects occur which are not covered by the model and need further investigation.

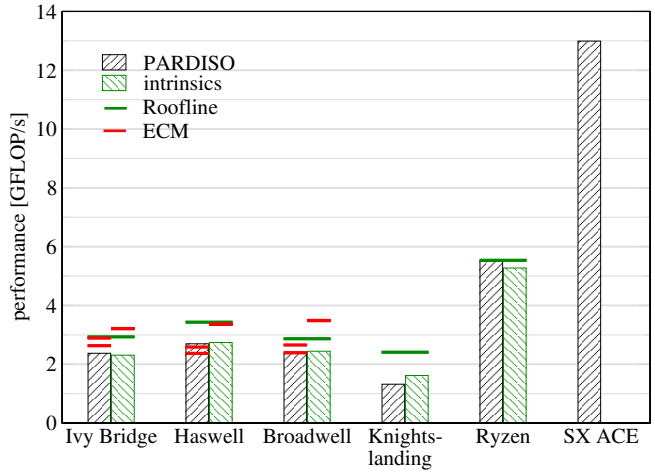

Figure 1: Measured performance and predictions of the Roofline and ECM model on one core of different hardware architectures for sparse triangular solve. The used factor with dimensions of $20\,000 \times 20\,000$ elements is dense with around 200×10^6 nonzero elements.

We would like to thank the High Performance Computing Center Stuttgart (HLRS) for providing access to the SX ACE machine and Holger Berger (NEC) for his technical support. This work was supported by the German Research Foundation (DFG) through the Priority Programme 1648 "Software for Exascale Computing" (SPPEXA) under WE-5289/1-2 project EXASTEEL-II.

References

[1] Schenk O., Gärtner K. Solving unsymmetric sparse systems of linear equations with PARDISO. Journal of Future Generation Computer Systems 2004; 20(3): 475–487. DOI:10.1016/j.future.2003.07.011.

[2] Williams S., Waterman A., Patterson D. Roofline: an insightful visual performance model for multicore architectures. Communications of the ACM 2009; 52(4): 65–76. DOI:10.1145/1498765.1498785.

[3] Hager G. et al. Exploring performance and power properties of modern multicore chips via simple machine models. Concurrency and Computation: Practice and Experience 2016. DOI:10.1002/cpe.3180.

[4] Treibig J., Hager G. Introducing a performance model for bandwidth-limited loop lernels. Proceedings of PPAM 2009. DOI:10.1007/978-3-642-14390-8_64.

Krylov–Schur type restarts for the two-sided Arnoldi method

Ian Zwaan[a] and Michiel E. Hochstenbach[b]

[a] *Faculty of Mathematics and Natural Sciences, University of Wuppertal*
[b] *Department of Mathematics and Computer Science, University of Eindhoven*

We consider the two-sided Arnoldi method applied to the unsymmetric eigenvalue problem and propose a Krylov–Schur type restarting method. We discuss the restart for regular Rayleigh–Ritz extraction as well as harmonic Rayleigh–Ritz extraction. Additionally, we investigate the convergence of the Ritz values and Ritz vectors and present generalizations of, e.g., the Bauer–Fike theorem and Saad's theorem. Applications of the two-sided Krylov–Schur method include the simultaneous computation of left and right eigenvectors and the computation of eigenvalue condition numbers. We also demonstrate how the method can be used to approximate pseudospectra and show that improvements in quality can be obtained over approximations with the (one-sided) Arnoldi method.

References

[1] A. Ruhe, *The two-sided Arnoldi algorithm for nonsymmetric eigenvalue problems*, in Matrix Pencils, B. K*r*agström and A. Ruhe, eds., Springer, 1983, pp. 104–120.

[2] G. W. Stewart, *A Krylov–Schur algorithm for large eigenproblems*, SIAM J. Matrix Anal. Appl., 23 (2001/02), pp. 601–614.

[3] G. W. Stewart, *Addendum to: "A Krylov–Schur algorithm for large eigenproblems"*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 599–601.