PROGRAMME AND ABSTRACTS

10th International Workshop on Parallel Matrix Algorithms and Applications (PMAA 18)

http://www.pmaa18.ethz.ch/

June 27 - 29, 2018

Venue:

CAB Building
Universitatstrasse 6

ETH zürich

PMAA18 Co-Chairs

Peter Arbenz (Switzerland), Rolf Krause (Switzerland), Daniel Kressner (Switzerland), and Olaf Schenk (Switzerland).

PMAA18 Programme committee:

Emmanuel Agullo (France), Pasqua D'Ambra (Italy), Hartwig Anzt (Germany), Haim Avron (Israel), Michael Bader (Germany), Achim Basermann (Germany), Costas Bekas (Switzerland), Paolo Bientinesi (Germany), Radim Blaheta (Czech Republic), Matthias Bollhöfer (Germany), Matthias Bolten (Germany), George Bosilca (USA), Edmond Chow (USA), Eric Darve (USA), Zlatko Drmac (Croatia), Nahid Emad (France), Efstratios Gallopoulos (Greece), Wilfried Gansterer (Austria), Pieter Ghysels (USA), Dominik Göddeke (Germany), Dan Gordon (Israel), Laura Grigori (France), Inge Gutheil (Germany), Hidehiko Hasegawa (Japan), Thomas Huckle (Germany), Julien Langou (USA), Hatem Ltaief (Saudi Arabia), Svetozar Margenov (Bulgaria), Karl Meerbergen (Belgium), Ulrike Meier-Yang (USA), Edoardo Di Napoli (Germany), Esmond G. Ng (USA), Maya Neytcheva (Sweden), Gabriel Oksa (Slovakia), Serge Petiton (France), Cosmin Petra (USA), Eric Polizzi (USA), Jean Roman (France), Jose E. Roman (Spain), Tetsuya Sakurai (Japan), Miroslav Tuma (Czech Republic), Marian Vajtersic (Austria), Wim Vanroose (Belgium), Kees Vuik (Netherlands), Weichung Wang (Taiwan), Roman Wyrzykowski (Poland).

Local organizing committee:

Peter Arbenz, Rolf Krause, Daniel Kressner, Olaf Schenk.

Dear participants,

Welcome to the 10th International Workshop on Parallel Matrix Algorithms and Applications (PMAA18)! The workshop co-chairs are happy to host this international conference here at ETH Zurich.

The PMAA workshop series aims to be a forum for an exchange of ideas, insights and experiences in different areas of parallel computing in which matrix algorithms are employed. The workshop will bring together experts and practitioners from diverse disciplines with a common interest in matrix computation.

The PMAA workshop series started in 2000 in Neuchâtel on the initiative of Erricos Kontoghiorghes as a tiny workshop. It grew and stabilized over the years. Two more workshops took place in Neuchâtel (2002, 2008), other venues were Marseille (2004), Rennes (2006), Basel (2010), London (2012), Lugano (2014), and Bordeaux (2016).

The PMAA'18 program consists of 4 plenary talks, 15 minisymposia sessions and 6 sessions of contributed talks. Altogether, there are around 90 talks.

The co-chairs tried hard to provide a balanced and stimulating programme that will appeal to the diverse interests of the participants. The local organizing committee hopes that the conference venue will provide the appropriate environment to enhance your contacts and to establish new ones. The conference is a collective effort of many individuals. The co-chairs, the scientific programme committee, and the local organizing committee have all contributed substantially to the organization of the workshop.

Peer reviewed papers presented at PMAA'18 will be considered for publication in a special issue of Elsevier's journal "Parallel Computing".

We all hope that you enjoy the workshop and your stay in Zurich.

The conference co-chairs:
Peter Arbenz (Switzerland)
Rolf Krause (USI Lugano)
Daniel Kressner (EPF Lausanne)
Olaf Schenk (USI Lugano)

The local organizers:
Peter Arbenz (ETH Zurich)
Daniel Hupp (ETH Zurich)

The special issue editors:
Olaf Schenk (USI Lugano), managing editor
Luc Giraud (Inria Bordeaux)
Wim Vanroose (University of Antwerp)
Peter Arbenz (ETH Zurich)

Contents

Schedule Overview

All lectures take place at CAB ETH Zürich, Universitätstrasse 6, CH-8092 Zürich, Switzerland.

Tuesday, 26th June 2018

05:00 PM - 08:00 PM Welcome Reception and Registration (CAB G Floor)

Wednesday, 27th June 2018

08:15 AM - 08:45 AM Registration (CAB G 11)

	CAB G 11	CAB G 61	CAB G 51
08:45 AM - 09:00 AM	Opening		
09:00 AM - 09:45 AM	IP 1.1 Scalable Eigensolver with Applications in Computational Physics and Chemistry (Chao Yang)		
09:45 AM - 10:15 AM	Coffee Break		
10:15 AM - 12:15 PM	MS 1.2.A Efficient dense eigensolvers - Methods and applications I	MS 1.2.B Scalable communication-reducing Krylov subspace methods	MS 1.2.C High performance accurate computing I
12:15 PM - 01:45 PM	Lunch		
01:45 PM - 03:45 PM	MS 1.3.A Efficient dense eigensolvers - Methods and applications II	MS 1.3.B Krylov and regularization methods for large scale inverse problems	MS 1.3.C High performance accurate computing II
03:45 PM - 04:15 PM	Coffee Break		
04:15 PM - 06:15 PM	CP 1.4.A Approximate and sparse factorizations	CP 1.4.B Multigrid and miscellaneous	CP 1.4.C Splitting methods

Thursday, 28th June 2018

	CAB G 11	CAB G 61	CAB G 51
08:50 AM - 09:00 AM	Notes from the organizers		
09:00 AM - 09:45 AM	IP 2.1 Domain Decomposition Methods: Theory and Applications (Frederic Nataf)		
09:45 AM - 10:15 AM	Coffee Break		
10:45 AM - 12:15 PM	MS 2.2.A Parallel eigenvalue solvers for large scale problems I	MS 2.2.B Parallelization aspects of SVD and EVD computations I	MS 2.2.C Task-based programming for scientific computing I
12:15 PM - 01:45 PM	Lunch		
01:45 PM - 03:45 PM	MS 2.3.A Parallel eigenvalue solvers for large scale problems II	MS 2.3.B Parallelization aspects of SVD and EVD computations II	MS 2.3.C Task-based programming for scientific computing II
03:45 PM - 04:15 PM	Coffee Break		
04:15 PM - 05:00 PM	IP 2.4 Scalable Tensor Algorithms for Scientific Computing (Edgar Solomonik)		
06:30 PM - 10:00 PM	Conference dinner		

Friday, 29th June 2018

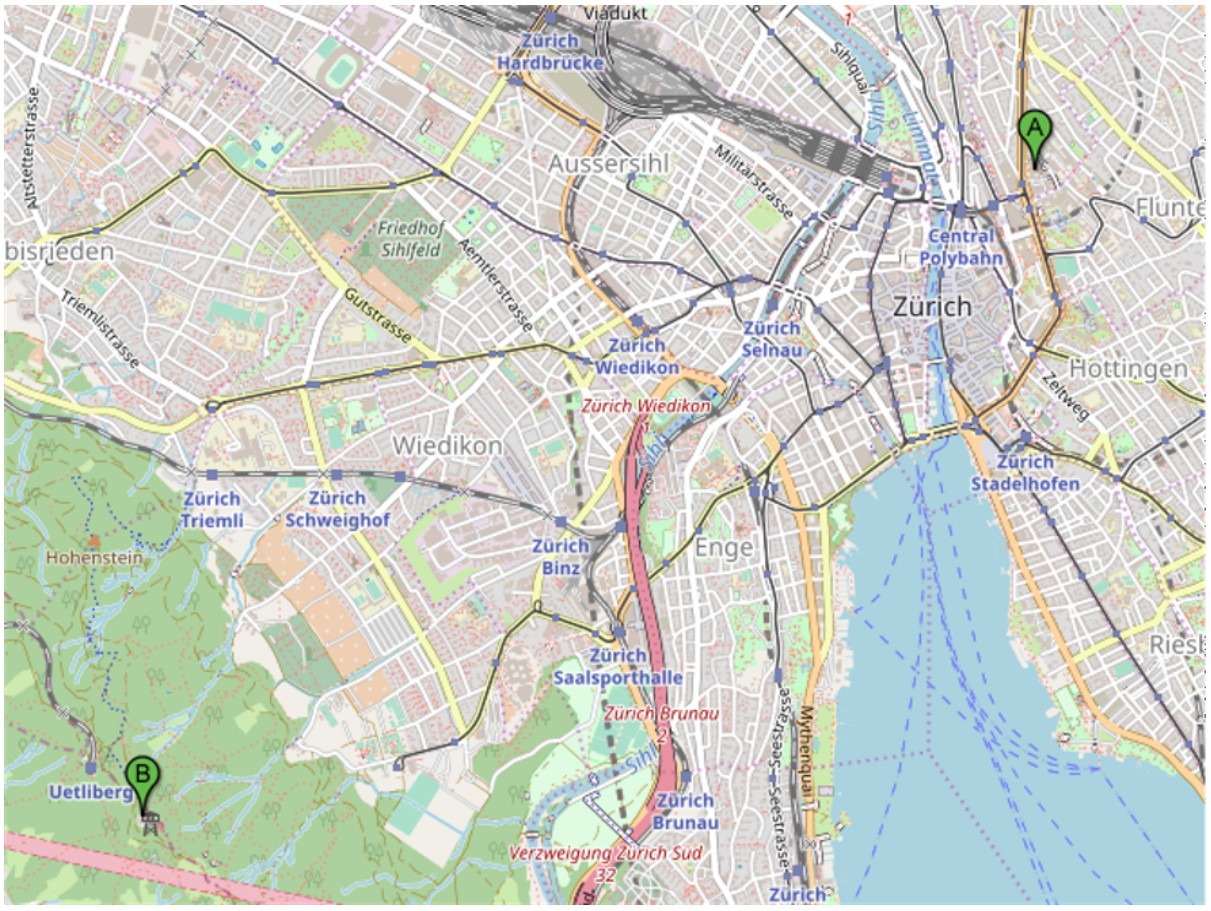
	CAB G 11	CAB G 61	CAB G 51
08:45 AM - 10:45 AM	CP 3.1.A Parallel and domain decomposition linear system solvers	CP 3.1.B Krylov space methods	CP 3.1.C Mixed precision and libraries
10:45 AM - 11:15 AM	Coffee Break		
11:15 AM - 01:15 PM	MS 3.2.A Recent advances in parallel sparse direct solvers	MS 3.2.B Parallel-in-time methods for HPC	MS 3.2.C Resilience in scientific computing
01:15 PM - 02:15 PM	Lunch on Site		
02:15 PM - 03:00 PM	IP 3.3 Does Machine Learning Need the Power of Iterative Methods for the SVD? (Andreas Stathopoulos)		
03:00 AM - 03:10 AM	Closing		

Social Events

- The coffee breaks will take place on the G-floor of the CAB building (see map on page VII).
- Welcome Reception (Tuesday 26th June, 5:00pm). The reception is open to all registrants. It will take place at the workshop venue in front of the lecture halls. The welcome reception gives you the opportunity to meet the other workshop attendees. It will be the first official event of the conference.
- Lunches are organised. On Wednesday and Thursday, the participants get a voucher for a lunch at the students' mensa (Polymensa). On Friday, there is a lunch on site.
- Conference Dinner (Thursday 28th June, 6:30pm). The conference dinner will take place in the restaurant of Hotel Uto Kulm (8143 Uetliberg / Zürich). The conference dinner is included in the conference registration fee. There is a fee for accompanying persons.

  You must have your conference badge in order to attend the conference dinner.

Important Locations



Conference Building: CAB Building, Universitätstrasse 6, 8092 Zürich

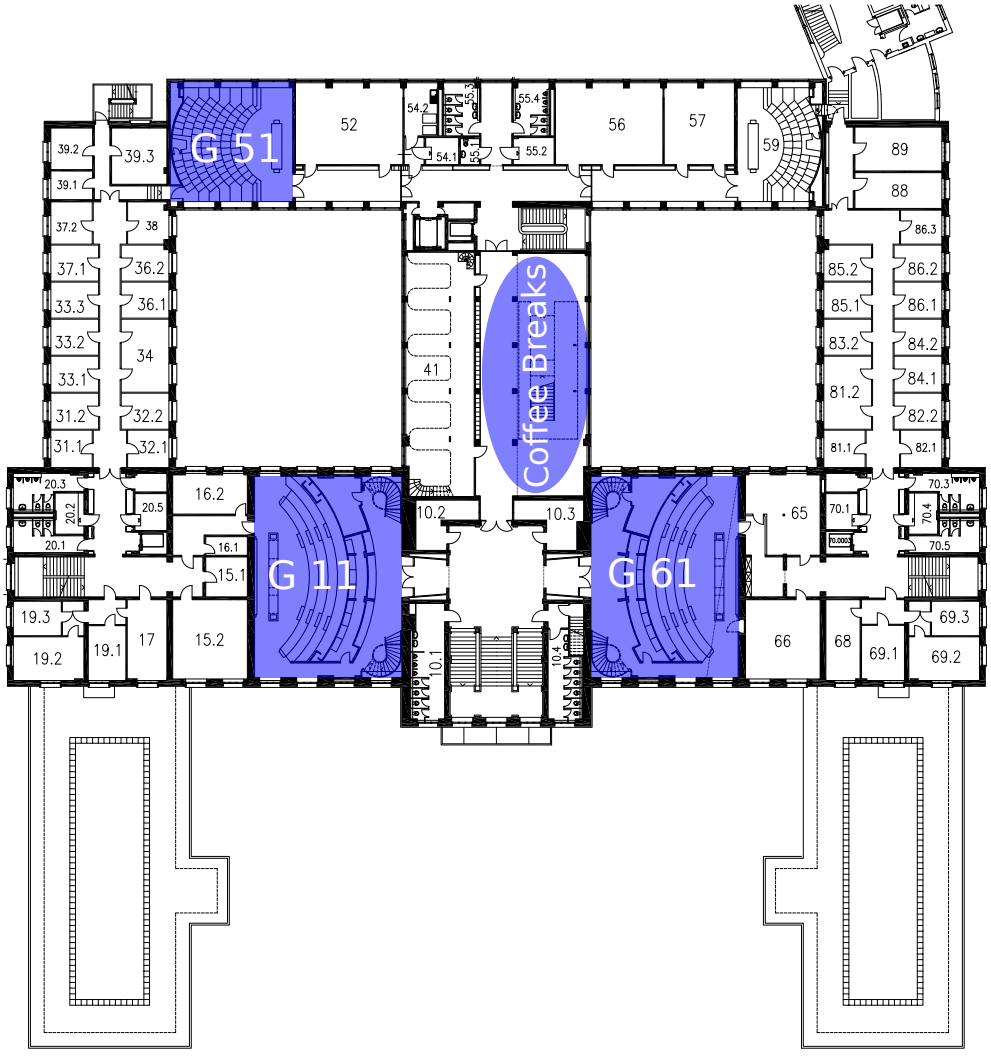
Conference Dinner: Hotel-Restaurant Uto Kulm, 8143 Uetliberg / Zürich

Directions

- From conference building to Zürich main station: Either walk (10-15 min) or take trams 10 or 6 to the main station.
- From Zürich main station to Uetliberg: Take the S10 train at the underground station on track 21/22 in direction Uetliberg. The train runs every half-hour at 17:35 and 18:05 and takes 27 minutes to get to the Uetliberg.

  From the final station at Uetliberg, there is a 7-minutes walk to the restaurant.

Conference Building



Plenary Talks (in chronological order)

Wednesday, 27.06.2018 9:00 AM - 9:45 AM Location: CAB G 11 Plenary talk 1.1

Scalable Eigensolver with Applications in Computational Physics and Chemistry

Speaker: **Chao Yang** Chair: Olaf Schenk

Solving the quantum many-body problem efficiently and accurately is one of the biggest challenges in computational physics and chemistry. There are broadly two approaches to seeking an approximate solution to this high-dimensional eigenvalue problem. One relies on projecting the many-body Hamiltonian onto a carefully chosen subspace of many-body basis functions. The other relies on constructing an effective mean-field model to capture the essential many-body physics that governs the interaction among different particles. These approaches yield algebraic eigenvalue problems that have different characteristics. Developing efficient computational schemes to tackle these problems on massively parallel computers requires choosing appropriate data structures to represent both the discretized Hamiltonian and the eigenvector to be computed, mapping such data structures onto a distribute memory multi-core processor grid, exploiting multiple threads within a computational node and improving the scalability of the computation by generating multiple levels of concurrency and reducing communication overhead. In this talk, I will give an overview on recent progress in these areas and point out the remaining challenges.

Thursday, 28.06.2018 9:00 AM - 9:45 AM Location: CAB G 11 Plenary talk 2.1

Domain Decomposition Methods: Theory and Applications

Speaker: **Frederic Nataf** Chair: Rolf Krause

Domain decomposition methods are a popular way to solve large linear systems on parallel architectures. These methods are based on a divide/conquer strategy. At each step of the algorithm, a problem is solved concurrently in each subdomain and then interfaces data are exchanged between neighboring subdomains. These are coarse grain algorithms since there are based on local volume computation and only surface data movements. Thanks to their very good ratio local computations/data movement, they are thus naturally well adapted to modern computer architectures. But, the original Schwarz method is slow. When implemented with a minimal overlap, it amounts to a block Jacobi method. Its convergence rate can be improved by using more generous overlaps and modifying the local blocks. In order to reach scalability, when the number of subdomains is large, a second level is introduced. At each step of the algorithm, a coarse problem with one or few unknowns per subdomain is used to further coordinate the solution between the subdomains globally. Theoretical results and numerical investigations (over a billion unknowns) for porous media flows, linear elasticity equations confirm robustness with respect to heterogeneous coefficients, automatic (non regular) partitions into subdomains and nearly incompressible behavior. Numerical results for large scale harmonic wave propagation phenomena will be shown. These results are obtained via an implementation in a Domain Specific Language devoted to the finite element method.

Thursday, 28.06.2018 04:15 PM - 05:00 PM Location: CAB G 11 Plenary talk 2.4

Scalable Tensor Algorithms for Scientific Computing

Speaker: **Edgar Solomonik** Chair: Daniel Kressner

Matrix and tensor eigenvalue computations consist of sequences of rectangular QR factorization and (sparse) tensor contractions (matrix products). We present results in improving the communication cost of these building blocks and show that they are optimal with respect to lower bounds. These algorithmic techniques show performance improvement across a range of parallel architectures. We make available distributed-memory implementations of these sparse and dense tensor algebra routines via the Cyclops library. We highlight the application of this C++/Python library to high-accuracy chemistry, quantum circuit simulation, and graph analysis.

Friday, 29.06.2018 02:15 PM - 03:00 PM Location: CAB G 11 Plenary talk 3.3

Does Machine Learning Need the Power of Iterative Methods for the SVD?

Speaker: **Andreas Stathopoulos** Chair: Peter Arbenz

Machine learning has emerged as one of the primary clients for large scale singular value calculations. Applications include clustering, recommendation systems, factor models in econometrics, and large-scale kernel methods. The matrices can be very large, although nonzero sparsity may not always be present. In some cases, one or two largest or smallest singular triplets are needed, while in other cases, a low rank approximation to the matrix is needed. One particular difference of these applications from traditional PDE applications is that low accuracy (typically 1-2 relative digits) is sufficient. To solve the SVD problem on large matrices, practitioners have traditionally turned to iterative methods such as Lanczos bidiagonalization or the restarted and preconditioned variants based on Davidson and LOBPCG. But for the specific requirements in machine learning, two different classes of methods are becoming increasingly popular. One class is the randomized SVD methods which focuses on choosing an appropriately sized initial space which with minimal iterations gives the desired space. The second class is streaming methods, where the matrix is accessed in its entirety but only once. In this talk we address the question of what problems are best suited for what type of methods, and present a unified view of randomized and iterative methods that is helpful both for developing and for using SVD software.

Parallel Sessions (in chronological order)

Wednesday, 27.06.2018 10:15 AM - 12:15 PM Parallel Session 1.2.

MS 1.2.A CAB G 11 EFFICIENT DENSE EIGENSOLVERS - METHODS AND APPLICATIONS I Chair: T. Huckle

#1: **Efficient reduction of dense HPD generalized eigenproblems to standard form**

Presenter: Valeriy Manin
Co-authors: Bruno Lang

We present an efficient approach to reduce a generalized eigenproblem $AC = BC\Lambda$ to a standard eigenproblem $\widetilde{A}\widetilde{C} = \widetilde{C}\Lambda$. Here A and B are hermitian and hermitian positive-definite, resp., matrices of size $N \times N$. Λ is a $K \times K$ diagonal matrix of the desired eigenvalues with $K \leq N$, and C is an $N \times K$ matrix of corresponding eigenvectors. We proceed with the reduction as follows:

[1] $B \Rightarrow U^H U$ (Cholesky decomposition of B; U is upper triangular, and U^H denotes the conjugate transpose of U)
[2] Optional: $U \Rightarrow U^{-1}$ (triangular matrix inversion)
[3] $A \Rightarrow \widetilde{A} = U^{-H} A U^{-1}$
[4] Solve the standard hermitian eigenproblem $\widetilde{A}\widetilde{C} = \widetilde{C}\Lambda$ for $\widetilde{C}$ and Λ
[5] $C = U^{-1}\widetilde{\Lambda}$ (back-transformation of the eigenvectors)

We concentrate on steps 3 and 5 and present highly scalable routines to implement matrix multiplications based on Cannon's algorithm. We show that our approach provides significantly better performance than the existing functions.

#2: **Efficient Transformation of the generalized Eigenproblem with symmetric banded matrices to a banded standard Eigenproblem**

Presenter: Michael Rippl
Co-authors:

The solution of symmetric eigenproblems plays a key role in many computational simulations. Generalized eigenproblems are transformed to a standard problem. This transformation has the drawback that for banded matrices in the generalized eigenproblem the banded structure is not preserved. The matrix of the standard eigenproblem will generally be a full matrix. We followed the ideas of the Group of Lang (University of Wuppertal) who modified Crawford's algorithm and implemented a procedure for small bandwidth to the ELPA project. By keeping the banded structure we save one reduction step on the matrix and one backtransformation step for the eigenvectors. This provides a good speedup compared to the standard tranformation procedure with Cholesky factorization.

#3: **Communication-Avoiding approaches of dense Eigenvalue / SVD problems**

Presenter: Toshiyuki Imamura
Co-authors:

For dense SEVP and SVD, the three types- of transformations based on the Householder transformation; tri-diagonalization, bi-diagonalization, and reduction to a Hessenberg form, are expensing tremendous communication cost in massively parallel processing. The communication avoiding (CA) and communication hiding (CH) approaches proposed for Householder tridiagonalization offer the significant cost reduction by removing 80% of 'the number of collective communications (startup overhead).' It reaches 15 to 20% of For dense SEVP and SVD, the three types- of transformations based on the Householder transformation; tri-diagonalization, bi-diagonalization, and reduction to a Hessenberg form, are expensing tremendous communication cost in massively parallel processing. The communication avoiding (CA) and communication hiding (CH) approaches proposed for Householder tridiagonalization offer the significant cost reduction by removing 80% of 'the number of collective communications (startup overhead).' It reaches 15 to 20% of the total computation time on a large scale parallel system such as K computer. For dense SEVP and SVD, the three types- of transformations based on the Householder transformation; tri-diagonalization, bi-diagonalization, and reduction to a Hessenberg form, are expensing tremendous communication cost in massively parallel processing. The communication avoiding (CA) and communication hiding (CH) approaches proposed for Householder tridiagonalization offer the significant cost reduction by removing 80% of 'the number of collective communications (startup overhead).' It reaches 15 to 20% of the total computation time on a large scale parallel system such as K computer. The principle of CA for Householder transformation consists reconstruct and reorder of the calculation of the reflector vector and matrix-product. Originally, we need two steps i) $u := a + \mathrm{sign}(a_1)|a|e_1$, then ii) $v := Au$. But, we apply that i) $[v', y] := A[a, e_1]$, ii) $\sigma := \mathrm{sign}(y_1)|a|, u := a + \sigma e_1, v := v' + \sigma y$. This reformation relaxes tight data dependency between the reflector u and obtaining v by the matrix product. Though it describes a one-side operation, it is also applicable to two-side operation. The similar idea of CA and CH can be applied to other two transformation methods. Proposal of two reconstructed transformation methods, especially the reduction to a Hessenberg form, its complexity analysis, and experimental results of K-computer are presented in the mini-symposium.

#4: **ELPA-AEO: recent optimizations for modern architectures**

Presenter: . . . . . Pavel Kus
Co-authors: . . . . . Hermann Lederer, Andreas Marek

The ELPA library is a well established eigensolver library used by many computational chemistry and materials science codes, which can be efficiently used for a large range of matrix sizes and on different hardware types with good scaling properties. To maintain its excellent performance on modern emerging architectures, a constant effort is required for various types of optimizations. In this contribution we present some of the recent results obtained within the ELPA-AEO project, concerning modern GPUs as well as modern Intel systems, including KNL and more recently the Skylake processors. An overview of recent hardware-specific optimizations will be given and performance comparisons will be shown.
Since there are multiple systems with different architectural features, each requiring slightly different approaches and different fine-tuning of many available parameters on the algorithmic level, it is increasingly difficult for the user to correctly select all the parameters in order to achieve the optimal performance for a given setup. It is also not possible to determine the optimal parameter setting for all combinations of problem and hardware setups a-priori, since there are simply too many of them. For this reason we introduced the autotuning capability within the ELPA library. It allows the user to automatically fine-tune the parameters, if the library is repeatedly used for similar problem setups, which is often the case in practical computations. We present an overview of this new feature and show examples of its practical application and possible performance benefits.
Part of the work is co-funded by BMBF grant 01IH15001.

MS 1.2.B CAB G 61 SCALABLE COMMUNICATION-REDUCING KRYLOV SUBSPACE METHODS Chair: S. Cools

#5: **Hiding global communication in the Conjugate Gradient method using deep pipelines**

Presenter: . . . . . Jeffrey Cornelis
Co-authors: . . . . . Siegfried Cools, Wim Vanroose

Krylov subspace methods are widely used as iterative solvers for sparse linear systems of equations. One of the most well-known algorithms is the Conjugate Gradient (CG) method by Hestenes and Stiefel [1]. Although the algorithm dates back to a paper published in 1952, the CG method is the main workhorse in lots of scientific applications, mainly because of its numerical simplicity and easy implementation.
Many of the Krylov subspace methods used today, including CG, were developed several decades ago, when the main goal was to minimize the total number of floating point operations (flops). However, with HPC hardware transitioning to the exascale regime, the current interest has shifted to increasing the parallel scalability of these iterative solvers on large parallel machines. The main bottleneck for scalability does not originate from the sparse matrix vector product (SPMV), since these often only require communication between neighbouring nodes, but from the calculation of dot-products. This is due to the fact that, after the local part of the dot-product has been calculated, the resulting number has to be communicated through a global reduction tree to gather the scalar result, after which this result has to be broadcast back again to each individual processor.
Two general ideas have emerged to improve parallel scalability; namely so-called “communication avoiding” and “communication hiding” methods. The first idea refers to reducing the total number of global reductions in the algorithm. In addition to avoiding synchronization bottlenecks, communication can be “hidden” by overlapping it with independent calculations. In pipelined Krylov subspace methods [2] the global reduction phase is overlapped with the SPMV computation, leading to improved parallel performance. However, when global reduction takes longer than the time required to compute an SPMV, communication cannot be overlapped completely. In this case it has been proposed to overlap the global reduction phase with the computation of multiple SPMVs, i.e. use a “deep pipeline”.
A pipelined variant of GMRES was developed by Ghysels et al. [2]. In analogy to the latter method, we have recently derived a variant of the Conjugate Gradient algorithm with deep pipelines [3], denoted as p(l)-CG. Although initial scaling results with various pipeline lengths are promising, numerical stability should be closely monitored [4]. Since the p(l)-CG algorithm uses several additional recurrence relations to update the approximate solution, the propagation of rounding errors in finite precision arithmetic typically differs from the classic CG algorithm.
This talk presents the basic theoretical ideas behind pipelined Krylov subspace methods with deep pipelines and comments on their parallel performance and numerical properties.

[1] Hestenes, M. R., Stiefel, E. (1952). Methods of conjugate gradients for solving linear systems . Journal of Research of the National Bureau of Standards, 49(6).
[2] Ghysels, P., Ashby, T. J., Meerbergen, K., Vanroose, W. (2013). Hiding global communication latency in the GMRES algorithm on massively parallel machines. SIAM Journal on Scientific Computing, 35(1), C48-C71.
[3] Cornelis, J., Cools, S., Vanroose, W. (2018). The communication-hiding Conjugate Gradient method with deep pipelines. submitted to SIAM Journal on Scientific Computing, arXiv preprint arXiv:1801.04728.
[4] Cools, S., Yetkin, E. F., Agullo, E., Giraud, L., Vanroose, W. (2018). Analyzing the effect of local rounding error propagation on the maximal attainable accuracy of the pipelined Conjugate Gradient method. SIAM Journal on Matrix Analysis and Applications, 39(1), 426-450.

#6: **Recycling Krylov method for the solution of sequence of linear systems**

Presenter: Hussam Al Daas
Co-authors: Laura Grigori, Pascal Hénon, Philippe Ricoux, Olivier Tisso

We propose a variant of GMRES to solve a sequence of linear systems issued from reservoirs simulations. It is based on enlarging the Krylov subspace, where multiple basis vectors are added at each iteration to enhance convergence. Since our method inherits a block scheme, we are interested in detecting inexact breakdowns. The matrix might not change from a linear system to the next in the sequence. It might also change but the difference is relatively small. Thus, we use recycling strategy of the Krylov subspace to improve the convergence. In its basic form, our method includes three global communication per iteration. We reorder mathematical operations in the basic method in order to reduce global communication. Scalable Communication-Reducing Krylov subspace methods.

#7: **Iteration-Fusing Conjugate Gradient**

Presenter: Sicong Zhuang
Co-authors: Marc Casas

We present the Iteration-Fusing Conjugate Gradient (IFCG) approach which is an evolution of the Conjugate Gradient method that consists in i) letting computations from different iterations to overlap between them and ii) splitting linear algebra kernels into subkernels to increase concurrency and relax data-dependencies. It presents two ways of applying the IFCG approach: The IFCG1 algorithm, which aims at hiding the cost of parallel reductions, and the IFCG2 algorithm, which aims at reducing idle time by starting computations as soon as possible. Both IFCG1 and IFCG2 algorithms are two complementary approaches aiming at increasing parallel performance. Extensive numerical experiments are conducted to compare the IFCG1 and IFCG2 numerical stability and performance against four state-of-the-art techniques. By considering a set of representative input matrices, we demonstrate that IFCG1 and IFCG2 provide parallel performance improvements up to 42.9% and 41.5% respectively and average improvements of 11.8% and 7.1% with respect to the best state-of-the-art techniques while keeping similar numerical stability properties. Also, we provides an evaluation of the IFCG algorithms' sensitivity to system noise and it demonstrates that they run 18.0% faster on average than the best state-of-the-art technique under realistic degrees of system noise.

#8: **Partial convergence in block Krylov solvers**

Presenter: Emmanuel Agullo
Co-authors: Luc Giraud, Yan-Fei Jing, Julien Langou, Thomas Mijeux

The solution of linear systems with multiple right-hand sides given simultaneously appear in many academic and industrial simulations. For large problem sizes, block variant of classical Krylov solvers appear as a suited solution technique to reduce the synchronization points but they must be equipped to properly manage the different convergence rate of the right-hand sides and possibly the convergence of linear combination of them that could lead to possible breakdowns or useless computation. In the talk, we will discuss numerical techniques that can be considered to handle such a situation; more precisely we will present two techniques in the context of block GMRES family and their implementation in the Fast Accurate Block Linear krylOv Solver (FABuLOuS) software package.

MS 1.2.C CAB G 51 HIGH PERFORMANCE ACCURATE COMPUTING I Chair: H. Hasegawa

#9: **High-performance implementations of reproducible and accurate matrix-multiplication**

Presenter: Daichi Mukunoki
Co-authors: Roman Iakymchuk, Stef Graillat, Takeshi Ogita

Parallel numerical computations with the underlying floating-point operations may suffer from round-off errors as well as their accumulations, which impact the accuracy and reproducibility of the final result. This can be observed not only for ill-conditioned, but also for regular problems. Thus, there is a need to guarantee accuracy and reproducibility of numerical computations. In this talk, we target to achieve both reproducible and correctly rounded matrix multiplication by applying the ExBLAS approach as well as the Ozaki solution. In addition, we employ some performance optimization techniques to obtain high-performance implementations on GPUs and provide the upper bound performance models for each implementation. Finally, we compare the obtained results against the double-precision cuBLAS implementation on various GPUs.

#10: **Reproducibility of sparse matrix-vector product and sparse solvers**

Presenter: Roman Iakymchuk
Co-authors: Daichi Mukunoki, Stef Graillat

Sparse systems of linear equations often arise in many applications in various domains from computational fluid dynamics to plasma physics and space weather forecast. Common approaches are the Jacobi method and the Conjugate Gradient method. In this talk, we aim to ensure reproducibility and accuracy of these two methods that could be violated in parallel executions due to, for example, the non-associativity of floating-point operations. Leveraging the hierarchical structure of linear algebra libraries, we construct our approach for these methods by

securing reproducibility and accuracy of their underlying building blocks such as sparse-matrix vector product (SpMV), dot product, and vector scaling. Finally, we present the performance and accuracy results for various GPUs.

#11: **Accurate Numerical Solutions of Large-Scale Linear Systems and Their Verification**

Presenter: Katsuhisa Ozaki
Co-authors: Takeshi Ogita

This talk concerns accurate numerical computations for large-scale linear systems. In a large matrix, rounding errors accumulate and lead to inaccurate computational results. We assume a dense matrix without any specified structure. We propose accurate numerical algorithms for the matrix-vector product based on [1,2] and implement them for parallel and distributed computers. Using the algorithms, a matrix-vector operation can be performed in a similar way to GEMV in BLAS and PBLAS as if computed in twice the working precision. Applying it to computation of residuals in iterative refinements, we can give accurate approximate solutions of linear systems. Additional computational cost for the iterative refinements is negligible compared to an LU decomposition of the coefficient matrix. Finally, we implemented two verification methods for linear systems [3,4] using PBLAS and ScaLAPACK. Verification methods aim to produce approximation solutions with their error bounds. Thanks to Fujitsu, in PBLAS we can safely use directed rounding such as roundTiesToEven, roundTowardPositive, roundTowardNegative, and roundTowardZero defined by IEEE 754 [5]. Applying accurate routines for a matrix-vector product to enclosure of residuals, we can obtain accurate approximate solutions with their tight error bounds. The presentation will show numerical examples using RIKEN's K computer.

[1] T. Ogita, S.M. Rump, and S. Oishi. Accurate sum and dot product. SIAM Journal on Scientific Computing, 26(6):1955–1988, 2005.
[2] N. Yamanaka, T. Ogita, S.M. Rump, and S. Oishi. A parallel algorithm for accurate dot product. Parallel Computing, 34(6–8):392–410, 2008.
[3] T. Ogita, S.M. Rump, and S. Oishi. Verified solution of linear systems without directed rounding. Technical Report 2005-04, Advanced Research Institute for Science and Engineering, Waseda University, Tokyo, Japan, 2005.
[4] S. Oishi and S.M. Rump. Fast verification of solutions of matrix equations. Numer. Math., 90(4):755–773, 2002.
[5] IEEE 2008. ANSI/IEEE 754-2008: IEEE Standard for Floating-Point Arithmetic. IEEE, New York.

#12: **Accurate eigenvector computations for clustered eigenvalues by iterative refinement**

Presenter: Takeshi Ogita
Co-authors: Kensuke Aishima

Efficient refinement Algorithms are proposed for symmetric eigenvalue problems. The structure of the algorithms is straightforward, primarily comprising matrix multiplications. Therefore, the algorithms benefit from highly optimized numerical libraries, such as BLAS, on parallel computers. We first present a basic algorithm to improve all eigenvectors associated with well-separated eigenvalues. We show that the proposed algorithm converges quadratically if a modestly accurate initial guess is given, including the case of multiple eigenvalues. Moreover, for multiple eigenvalues, we prove quadratic convergence whenever all simple eigenvalues are well separated. The convergence rate is also preserved in finite precision arithmetic if the working precision is sufficiently high in the algorithm. Our convergence analysis can be extended to Hermitian matrices. On the basis of the basic algorithm, we propose a practical algorithm that can improve eigenvectors associated with clustered eigenvalues. Iterative use of the proposed algorithms computes an eigenvalue decomposition of a real symmetric matrix that is accurate according to working precision. Numerical results demonstrate excellent performance of the proposed algorithm in terms of convergence rate and overall computational cost, and show that the proposed algorithm is considerably faster than a standard approach using multiple-precision arithmetic.

Wednesday, 27.06.2018 01:45 PM - 03:45 PM Parallel Session 1.3.

MS 1.3.A CAB G 11 EFFICIENT DENSE EIGENSOLVERS - METHODS AND APPLICATIONS II Chair: B. Lang

#13: **Open Infrastructure for Large-Scale Kohn-Sham Density-Functional Theory: The ELSI Project**

Presenter: . . . . . Victor Yu
Co-authors: . Alberto Garcia, William Huhn, Mathias Jacquelin, Weile Jia, Murat Keceli, Raul Laasner, Yingzhou Li, Lin Lin, Jianfeng Lu, Alvaro Vazquez-Mayagoita, Chao Yang, Haizhao Yang, Volker Blum

Molecular and materials simulations based on Kohn-Sham density-functional theory (KS-DFT) are the production workhorses for a broad range of applications in physics, chemistry, biology, and materials science. In large-scale KS-DFT calculations, solving or circumventing a generalized eigenvalue problem is often the major computational bottleneck, which limits the achievable system size to roughly several thousand atoms. This is a generic problem that must be addressed by essentially all current electronic structure codes. Conventional KS-DFT implementations solve the eigenvalue problem by direct or iterative diagonalization, whose computational complexity scales cubically with respect to the system size. There exist alternative algorithms that circumvent the explicit solution of the eigenvalue problem by directly computing the density matrix. With a smaller scaling exponent and a larger prefactor, these methods can potentially outperform diagonalization for large systems beyond a thousand atoms. We here present an open-source, integrated software interface, ELSI (`http://elsi-interchange.org`), to simplify the access to existing strategies to address the KS eigenvalue problem for different problem classes on different scales. Currently supported algorithms are the massively parallel dense eigensolver ELPA, the orbital minimization method implemented in libOMM, the pole expansion and selected inversion method (PEXSI), and the shift-and-invert parallel spectral transformation eigensolver (SIPS), including both cubic scaling and reduced scaling methods. The ELSI interface aims to simplify the implementation and optimal use of these methods, by providing (a) reasonable default settings for a chosen solver, (b) automatic conversion between input and internal working matrix formats, and (c) suggestions on the optimal solver for a given problem. Comparative benchmarks performed on distributed memory supercomputing architectures are presented for system sizes up to ten thousand atoms. The strengths and limitations of the solvers will be discussed. Finally, we introduce our proposal of constructing a meta-benchmark set and using it to facilitate the development and test of existing and new solvers in electronic structure theory.

#14: **Performance benchmark of standard eigensolver on KNL systems**

Presenter: . . . . . Inge Gutheil
Co-authors: . . . . . .

With the invention of many-core systems like the Intel KNL standard eigensolver libraries have to be adapted to those architectures. The pure MPI parallelizing strategy may be no longer suited for these new architectures because too many MPI processes need too much memory for buffers.
In this talk we will present the first performance evaluation results of the eigensolver libraries ELPA and EigenExa on the JURECA booster KNL nodes. Both libraries are tuned for KNL usage and they offer a hybrid parallelization with MPI in combination with OpenMP. The ELPA 2-step eigensolver provides special kernels with KNL intrinsics for the back transformation of eigenvectors and their usage indeed leads to better performance on KNL than using just AVX2 kernels.
On a single KNL node still the pure MPI versions of both libraries deliver the best performance, but when more nodes are used the hybrid parallelization becomes superior.
Up to now the JURECA booster module is still a standalone system but in future it will be integrated to the Haswell system JURECA to allow applications that use both parts in combination. In order to see whether it makes sense to offload the solution of an eigenproblem we investigate the performance of the libary eigensolvers on the KNL nodes now and compare it to the performance of the same routines on the Haswell nodes.

#15: **Eigenvalue problems in large-scale first-principles electronic structure calculations**

Presenter: . . . . . Jun-Ichi Iwata
Co-authors: . . . . . .

First-principles electronic structure calculation based on the Density Functional Theory (DFT) has been an indispensable tool for many fields of materials science and engineering. With the development of supercomputers, the target size of first-principles DFT calculations becomes larger and larger, and nowadays, a few hundreds to a thousand of atoms has been computable with standard plane-wave based DFT program codes. However, the computable sizes are still not satisfactory for clarifying or designing the material properties in realistic situations. The challenge for large scale calculations with state-of-the-art supercomputers is one of the ways to overcome the size difficulty in the first-principles electronic structure calculations. In this talk, I'd like to introduce our program code RSDFT, which has been developed to perform large-scale first-principles calculations on massively-parallel computers including the Japanese flagship machine K computer. RSDFT is based on the real-space finite-difference pseudopotential method. The basic equation of RSDFT is a discretized Kohn-Sham equation, which is a nonlinear eigenvalue problem with a large-sparse matrix. In the RSDFT code, we solve

the nonlinear eigenvalue problem with a subspace-iteration method combining a Broyden's or Pulay's method for nonlinear equations. Contrary to the standard plane-wave methods, the real-space method needs not to use Fast Fourier Transformations, which requires heavy communication burden, and therefore the scalability of RSDFT is rather good even in the calculations with tens of thousands of compute nodes. It has also been started to develop RSDFT for the next flagship computer called post-K computer, and we aim to make first-principles calculations on the system with a few thousand of atoms easy tasks. I would like to also talk about the development of RSDFT for the post-K computer.

#16: **Ab initio Materials Simulations: A Challenge for Eigensolvers**

Presenter: Christian Carbogno
Co-authors: Matthias Scheffler

Over the last decades, the development of *ab initio* electronic structure theory and especially density-functional theory (DFT) has enabled unpreceded insights and advancements in chemistry, solid state theory, and material science. However, the N^3 scaling of matrix diagonalizations is the computational bottleneck in today's DFT calculations, which hinders the systematic investigation of large system sizes ($> 1{,}000$ atoms). Addressing such system sizes is of exceptional importance to solve many grand challenges of scientific, societal and industrial relevance.
For electronic structure theory applications, ELPA is the leading library for the massively parallel, highly scalable direct solution of eigenproblems. In this talk, we discuss how recent advancements in ELPA can now be utilized in the electronic-structure code FHI-aims to have an impact at the application level, so to enable *faster* calculations for *larger* system sizes. In particular, we discuss how this facilitates running *ab initio* molecular dynamics simulations that are necessary for the accurate and reliable assessment of thermal conductivities.
* *This work has been performed together with the ELPA-AEO consortium (`http://elpa-aeo.mpcdf.mpg.de`) within the ELPA-AEO project (BMBF 01IH15001)*

MS 1.3.B CAB G 61 KRYLOV AND REGULARIZATION METHODS FOR LARGE SCALE INVERSE PROBLEMS

Chair: W. Vanroose

#17: **Krylov methods for the Helmholtz equation in forward and inverse problems.**

Presenter: Wim Vanroose
Co-authors:

Krylov methods for Helmholtz and Scattering equation
Inverse and forward scattering problems require the repeated solution of a Helmholtz problem. These appear in imaging, chemistry, acoustic and many other applications.
Over the years various iterative Krylov methods with special preconditions have been developed. Such as complex shifted Laplacian, wave ray or sweeping preconditioners.
However, within the framework of inverse and forward scattering problems the observable is not necessarily the solution of the Helmholtz equation but rather an integral over the solution. Indeed, for example the far field map is calculated as a volume or, through stokes, a surface integral over the solution. This leaves additional freedom to develop an efficient solver since the solution of the Helmholtz equation is only an intermediate result.
In this talk we illustrate that if we deform the contour of integration, we have to solve a complex shifted Helmholtz problem rather then a Helmholtz equation with a real wavenumber. The first is easy to solve with multigrid.
The underlying intuition is that the observables are integrals over rapidly oscillating functions. Why spend a lot of effort of solving the Helmholtz equation accurately to resolve all the oscillations if they are cancelled out anyway by the integral? The integral is determined by the critical points of the oscillation. They need to be represented well and this can be done by a complex shifted problem.

#18: **Generalized Davidson and multidirectional-type methods for the GSVD**

Presenter: Ian Zwaan
Co-authors: Michiel Hochstenbach

We propose two new iterative methods for computing nontrivial extremal generalized singular values and vectors. The first method is a generalized Davidson-type algorithm and the second method employs a multidirectional subspace expansion technique. Both methods allow for tick restarts. Essential to the latter method is a fast truncation step designed to remove a low quality search direction and to ensure moderate growth of the search space. Numerical experiments indicate that both methods are competitive.

#19: **Parallel algorithms for hyperbolic PDE-constrained optimization problems**

Presenter: Andreas Mang
Co-authors: George Biros, Amir Gholami

We present effective algorithms for the solution of hyperbolic PDE-constrained optimization problems with applications in medical image analysis. Our contributions are: (i) we examine algorithmic scalability of our memory-distributed solver; (ii) we present and study an improved implementation of the computational kernels of our solver (fast Fourier transform and cubic interpolation) of our solver; (iii) we explore different variants of the preconditioner

for the reduced space Hessian; and (iv) we report results for the performance of our methods on clinically relevant problems.
We use a globalized, matrix-free Newton–Krylov method for numerical optimization. We use a spectral collocation scheme for the discretization in space, and an unconditionally stable semi-Lagrangian scheme for the integration in time. Our code is implemented in C++ and uses the message passing interface (MPI) library for parallelism. We will study the rate of convergence, time-to-solution, and inversion accuracy of our solver. We will report scalability results for different high-performance computing platforms. We will see that our distributed-memory solver allows us to solve problems of unprecedented scale (with up to 200 billion unknowns). We will see that our improved solver yields a speedup of up to one order of magnitude compared to the state-of-the-art; we can solve clinically relevant problems in less than 2 minutes on one node with 24 cores.

#20: **Projected Newton method for a system of Tikhonov-Morozov equations**

Presenter: Nick Schenkels
Co-authors: Wim Vanroose

Many inverse problems can be written as sparse, large scale linear or nonlinear systems. Newton-Krylov subspace methods are well suited to deal with these kinds of problems and there exists a vast literature on how these methods can be efficiently implemented on large parallel systems. However, in order to solve an inverse problem some form of regularization is typically required. Often this results in a Tikhonov-like formulation of the problem where minimizing the discrepancy between the model predictions and the observed data is balanced with minimizing a regularization term. This balance is governed by a so-called regularization parameter that has to be determined, which is a non-trivial issue. In many applications this regularization parameter is chosen by trial-and-error or by using some grid based approach, which are both inefficient and computationally expensive. It is, however, possible to write down a non-linear system of equations for both the solution of the inverse problem and the regularization parameter by combining the Tikhonov normal equations and Morozov's discrepancy principle.
If this system is solved using Newton's method, convergence can in general not be guaranteed. We therefore derive a limit on the Newton step sizes and prove that starting from a point that satisfies the Tikhonov normal equations for any regularization parameter we can guarantee the convergence. Because each Newton iteration requires, amongst others, solving the Jacobian system for the Newton search direction, this method is computationally expensive – even for small inverse problems. By using a bidiagonal decomposition of the matrix it is, however, possible to project the non-linear system onto a low-dimensional Krylov subspace where this is no longer an issue. We also present numerical results from applications to benchmark matrices and computed tomography that illustrate the workings of these methods and compare them with other known regularization methods.

MS 1.3.C CAB G 51 HIGH PERFORMANCE ACCURATE COMPUTING II Chair: H. Hasegawa

#21: **A fast and efficient preconditioning method for solving ill-conditioned dense linear systems using partly simplified LU factors**

Presenter: Yuka Kobayashi
Co-authors: Takeshi Ogita

We consider solving an ill-conditioned dense linear system

$$Ax = b, \quad A \in \mathbb{R}^{n\times n}, \quad b \in \mathbb{R}^n, \tag{1}$$

where the condition number $\kappa(A)$ is too large to solve (1) by ordinary an floating-point arithmethic. We propose a preconditioning method to obtain an accurate approximate solution $\tilde{x}$ of (1). The method is based on the previous preconditioning method [1] using an LU factor.
Let u denote the relative rounding error unit in working precision of floating-point arithmetic. If $\kappa(A)$ becomes large such that $u \cdot \kappa(A) \geq 1$, an approximate solution of (1) becomes unstable. Moreover, in such case, iterative refinement methods using LU factors cannot work. To overcome this, a possibility is to use multiple-precision arithmetic. However, if we apply multiple-precision arithmetic to entire computations, computing time increases significantly regardless of the size of condition number of A. To remedy these defects, if A is ill-conditioned, we apply preconditioning methods to (1) as

$$MAx = Mb, \quad M \in \mathbb{R}^{n\times n}$$

for reducing the condition number of A such that $\kappa(MA) \ll \kappa(A)$.
In the previous preconditioning method in [2], an approximate inverse of A is adopted as M for problems such that $\kappa(A) \leq (u^{-1})^2$. In a similar way, in [1], an LU factor is adopted as M to reduce computational cost. If we use Crout's LU factorization for A, then it is likely that $\kappa(A) \approx \kappa(L)$. We can utilize this nature to decrease the condition number of A with a left preconditioner as follows. First, we execute Crout's LU factorization of A' such that $A' = PA \approx LU$. Next, we obtain $X_L \approx L^{-1}$ and $b' = Pb$. Then, we have

$$X_L A' x = X_L b'.$$

Here, it is expected that $\kappa(X_L A') \approx 1 + u \cdot \kappa(A)$.
In this presentation, we focus on the distribution of singular values. We define the singular values of A such that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$. Suppose that A has m relatively large singular values as

$$\sigma_m \geq \alpha \cdot \sigma_1, \quad \sigma_{m+1} < \alpha \cdot \sigma_1,$$

where α is the threshold with $u \leq \alpha \leq 1$. Using Crout's LU factorization for A, the structure of L is likely to be as follows.

$$L = \left[\begin{array}{c|c} L_{11} & O \\ \hline L_{21} & L_{22} \end{array}\right] \begin{matrix} \}m \\ \}n \end{matrix}$$

$$|(L_{11})_{ii}| \geq \alpha \|A\|_\infty \quad (i = 1, \cdots, m)$$
$$|(L_{22})_{11}| < \alpha \|A\|_\infty$$

We observe that the magnitude of L_{22} is relatively too small to be useful, and replacing L_{22} with αI_{n-m} does not affect the effect of preconditioning where I is the identity matrix. Therefore, we decide to replace L with

$$\tilde{L} := \left[\begin{array}{cc} L_{11} & O \\ L_{21} & \alpha I_{n-m} \end{array}\right].$$

After that, we compute $X_L \approx \tilde{L}^{-1}$ and solve $X_L A x = X_L b$. Then it is expected that $\kappa(X_L A) \approx 1 + \alpha\kappa(A)$. The proposed method can significantly reduce computational cost for preconditioning. We will show the numerical results in our presentation.

[1] Y. Kobayashi, T. Ogita, Accurate and efficient algorithm for solving ill-conditioned linear systems by preconditioning methods, NOLTA, IEICE, 7 (2016), 374–385

[2] S. M. Rump, Approximate inverses of almost singular matrices still contain useful information, Forschungsschwerpunktes Informations- und Kommunikationstechnik, Technical Report 90.1, Hamburg University of Technology, Hamburg, Germany, 1990.

#22: **Strategy of Precision Switching for Mixed Precision Iterative Method**

Presenter: Masaki Suwa
Co-authors: Akihiro Fujii, Teruo Tanaka, Hidehiko Hasegawa

The convergence of Krylov subspace methods is influenced by the rounding errors. The high precision arithmetic operation improves the convergence, however the computation time increased. The mixed precision iterative method proposed by Kotakemori et al., called SWITCH, is one of the methods to reduce computation time. SWITCH solves partway in double precision and restarts its iterative process with high precision using the result in double precision as an initial solution. As in this method, the mixed precision iterative method is speeded up by not using the high precision arithmetic operation as much as possible. Kotakemori et al. evaluated the mixed precision iterative method using double precision and double-double precision. However, the required precision is matrix dependent and there are cases which cannot be solved even with double-double precision. We evaluated SWITCH type mixed precision iterative method with more flexible and high precision using GMP. Especially, we tested GMRES, GCR, and BICGSTAB iterative methods. In this presentation, we examine the effectiveness of precision switching strategy for the SWITCH type mixed precision iterative method using arbitrary precision.

#23: **Generation of large scale matrices for numerical examples**

Presenter: Takeshi Terao
Co-authors: Katsuhisa Ozaki

A test matrix is useful for checking the accuracy and stability of algorithms in numerical linear algebra. Our aim in the present study is to develop efficient algorithms that produce very large-scale dense matrices with specified singular values for parallel and distributed computers. The target dimension of such matrices is greater than 1,000,000. MATLAB's built-in 'gallery' function can set five distributions of singular values: one large singular value, one small singular value, geometrically distributed singular values (default), arithmetically distributed singular values, and random singular values with uniformly distributed logarithm. The aim is to implement such functions on parallel and distributed computers. First, two approximate orthogonal matrices and a diagonal matrix are generated. Next, we multiply the orthogonal matrices from the left and the right sides with the diagonal matrix. The discussion is based on a form of singular value decomposition. One possibility for obtaining the approximate orthogonal matrices is to apply QR decomposition, that is, decomposition into a product QR of an orthogonal matrix Q and an upper triangular matrix R. We implemented two methods for generating the test matrix by using the specified singular values based on PBLAS and Scalapack. One is based on QR decomposition (pdgeqrf). The other is based on ChoekskyQR and its extension (called ChoelskyQR2). We will compare the computing times and the accuracies of the two methods for shared memory computers. In addition, efficiency of parallelization for a large scale matrix on the Riken K computer will be introduced.

#24: **Accurate Interval Matrix Computations**

Presenter: . . . . . Stef Graillat
Co-authors: . . . . .

In this talk, we will present some accurate and parallel algorithms for interval computations with matrices.

Wednesday, 27.06.2018 04:15 PM - 06:15 PM Parallel Session 1.4.

CP 1.4.A CAB G 11 APPROXIMATE AND SPARSE FACTORIZATIONS Chair: M. Bollhöfer

#25: **An improved exact algorithm and an NP-completeness proof for 2D sparse matrix partitioning.**

Presenter: Timon Knigge
Co-authors: Rob Bisseling

Sparse matrix-vector multiplication (SpMV) is a common elementary operation performed in numerical algorithms. In applications involving very large matrices, this operation can be accelerated considerably by distributing the computation over multiple processors, giving each of k processors a subset of roughly $\frac{N}{k}$ out of N nonzeros. When constructing such a balanced partitioning we typically optimize for minimal total communication volume between processors during fanout/fanin phases. Unfortunately, sparse matrix partitioning is a computationally hard problem so that in practice only heuristic solutions are useable. Still, exact algorithms are useful to generate benchmarks and reveal where heuristics succeed or fail. We build on recent work by Pelt and Bisseling (2015) who proposed an exact combinatorial branch-and-bound solution, extending it with tighter bounds to solve larger instances. We also give a reduction showing that the 2D sparse matrix partitioning problem is $\mathcal{NP}$-Complete even when the number of processors is fixed to $k = 2$.

#26: **Solving ill-conditionned linear systems using extended sparsification: an application to extruded meshes**

Presenter: Leopold Cambier
Co-authors: Chao Chen, Siva Rajamanickam, Erik Boman, Raymond Tuminaro, Eric Darve

Solving highly ill-conditioned sparse linear systems remains a challenging task in scientific computing. In this project, we explore the application of the extended sparsification method on a matrix arising from the discretization of an elliptic PDE on an extruded mesh.
The extended sparsification algorithm is a technique where one compresses fill-in edges arising from far fields interactions and introduces new unknowns to preserve the sparsity pattern of the original linear system. This results in a generic (parallel) algorithm with little customization required, except the partitioning algorithm and the compression tolerance parameter. It can be used either as a low-accuracy direct solver or a preconditioner coupled with an iterative method. The low-rank basis can be further improved by adding custom vectors provided by the user. If those correspond to the small eigenvalues of the system, it typically improves the accuracy of the solver.
The studied application comes from the discretization of ice-flows on Antarctica. Because the physical quantities of interest have large variations in scale, the solution is very sensitive to small scales and the overall system has a condition number of more than 10^{10}.
This property of the system happened to severely affects the ability to use the original solver as a black-box preconditioner. Without any modifications, the algorithm performs poorly, needing > 100 GMRES iterations on even low-resolution problems. With a geometry-aware domain partitioning, the solver works better. However, the number of iterations still does not decrease until one reaches high accuracies, hinting at the fact that most of the relevant information is hidden in the small scales.
The key in making the algorithm efficient is to scale yet-to-be-compressed edges with the diagonal pivot. This usually has little effect on moderately ill-conditioned problems; however, since in this case some pivots may be near-singular, the scaling has a significant effect. With this modification, the preconditioner behaves significantly better: the number of iterations steadily decreases starting at low accuracies.
Finally, we apply the algorithm on problems having from 60 000 to 500 000 000 unknowns. We successfully demonstrate that the algorithm is efficient, scaling much better than direct methods. It is also more versatile that for instance multigrid and can be applied to a wide range of problems with little to no modification.

#27: **High Performance Large-Scale Matrix Inversion using Block Incomplete LU Factorizations**

Presenter: Matthias Bollhöfer
Co-authors: Olaf Schenk

We consider application problems that heavily rely on computing parts of the matrix inverse approximately. Among these problems are large sparse inverse covariance matrix estimation as well selective inversion in electronic structure calculations. Typically the accuracy that is required to compute parts of these matrix inverses is relatively high, thus using incomplete factorizations and inverting the triangular factors leads to relatively dense matrices which makes usual incomplete factorization infeasible for this kind of applications. Instead we propose the use of block incomplete factorization methods where appropriate block structures are constructed to allow for the use of dense matrix kernels. We demonstrate the use of these approximate factorization methods inside applications that require matrix inversion. Furthermore, the benefits of parallelized matrix inversion and the use of multi-threaded dense matrix kernels is demonstrated.

CP 1.4.B CAB G 61 MULTIGRID AND MISCELLANEOUS Chair: E. Gallopoulos

#28: **Performance and implementation of a geometric multigrid solver with Trilinos**

Presenter: Matthias Frey
Co-authors: Andreas Adelmann

The Poisson problem arising from large-scale N-body problems coupled with Maxwell's equations in the electrostatic limit represents an accuracy and efficiency bottleneck in simulations of neighbouring bunch effects in high intensity cyclotrons. Standard particle-in-cell models are not able to capture the tiny effects between bunches without wasting memory in regions of void due to the uniformity of the fine mesh. Block-structured adaptive mesh refinement algorithms are a suitable method to overcome this issue. Their hierarchy of levels and grids is applied to solve Poisson's equation using an adaptive geometric multgrid algorithm. This talk presents a new implementation of Martin's and Cartwright's algorithm based on Trilinos. Furthermore, a benchmark study of various preconditioners and solvers is shown with a comparison to AMReX's multigrid solver. A scalability test up to 13'824 cores shows a parallel efficiency of around 60% on Piz Daint.

#29: **AMG based on compatible weighted matching for GPUs**

Presenter: Dario Pasquini
Co-authors: Massimo Bernaschi, Pasqua D'Ambra

We describe the main issues found in the design of an efficient implementation, tailored to GPGPUs, of an Algebraic MultiGrid (AMG) preconditioner recently proposed by one of the authors and already available for CPU in the open-source BootCMatch code. The AMG method relies on a new approach for coarsening sparse symmetric positive definite matrices, which we refer as coarsening based on compatible weighted matching. It exploits maximum weight matching in the adjacency graph of the sparse matrix and the principle of compatible relaxation to define a pairwise aggregation of unknowns. The aggregates are unknown pairs coupled in a maximum product matching of the original sparse matrix graph with suitable edge weights. The final aim is to enhance the diagonal dominance of a matrix that is the hierarchical complement of the resulting coarse matrix with respect to a given scalar product, thereby improving the convergence properties of a corresponding compatible relaxation scheme. The matched nodes are aggregated to form coarse set of unknowns, and piecewise constant or smoothed interpolation operators are applied for the construction of a multigrid hierarchy. No reference is made to any a priori knowledge on the matrix origin and possible information about smooth errors are used to define edge weights assigned to the original matrix graph. More large aggregates can be obtained by combining multiple steps of the basic pairwise aggregation. The most demanding kernel in this type of coarsening is the computation of an efficient maximum product matching. Accurate solutions for computation of maximum product matching in a graph are based on the Hungarian algorithm to search optimal augmenting paths in the matrix between unmatched vertices. This algorithm is a sequential process representing a roadblock in the search for an efficient parallel computation of a maximum weight matching. In our attempt to exploit high computing power of GPUs in the design of a parallel coarsening based on compatible weighted matching, we adopt an approximate solution widely used in related fields, such as in coarsening strategies for multilevel data partitioning as well as in scaling and permuting sparse matrices for efficient parallel direct solvers. We show that approximate solutions based on a recently proposed multithreaded matching algorithm, referred as the suitor algorithm, allow us to obtain good quality coarse matrices for our AMG on GPUs, largely reducing run times with respect to the original sequential algorithm implemented in BootCMatch. We will show results on a large set of sparse matrices arising from discretization of partial differential equations as well as from Laplacian operators of general complex graphs.

#30: **Adapting MPRGP algorithm for solving SVM problems using PermonSVM**

Presenter: Marek Pecha
Co-authors: Martin Cermak, Zdenek Dostal, Vaclav Hapla, David Horak, Jakub Kruzik

The presentation deals with adapting quadratic programming (QP) algorithms implemented in our PERMON toolbox, namely its PermonQP module, for machine learning problems of the Support Vector Machines (SVM) type.
PermonSVM is a new SVM tool designed to run in parallel. It is written on top of PETSc and PermonQP, which are parallelized using MPI. The parallelism comes mainly from the distribution of matrices across processes. PermonSVM provides an implementation of classifications via soft-margin SVM with the linear kernel. In the training procedure, PermonSVM takes advantage of the scalable matrix-vector product in PETSc and an implicit representation of the Hessian matrix, which saves memory and CPU time. Additional features include a probability SVM output, fast, load-balanced cross-validation and grid search for parameter tuning, L1 and L2 hinge-loss functions, and parallel LIBSVM and HDF5 file loader. PermonSVM provides an executable for SVM classification as well as PETSc-like C API.
Our team has a long-term experience with the development of massively parallel and scalable implementations of QP algorithms especially for problems arising in contact mechanics in combination with FETI domain decomposition methods. Driven by this know-how, we have tried to apply the most successful solvers, namely MPRGP (Modified Proportioning and Reduced Gradient Projection) and its modifications to the solution of QP problems arising in SVM. MPRGP, designed by Dostal et al., is an efficient algorithm for the solution of convex QP with box

constraints.
MPRGP is an active set method. Its basic version can be considered as a modification of the Polyak algorithm. MPRGP combines the proportioning algorithm with the gradient projections. One of the ingredients of the algorithm is the expansion of the active set using fixed step size. The convergence rate is proven for the expansion step size bounded by twice the reciprocal value of the Hessian norm. Experiments show that this step size is too short for SVM problems, resulting in a huge number of expansion steps. We will present an adaptive expansion step size that reduces the number of expansion steps by a factor of up to seven. Further, other numerical results computed with different hinge loss functions, a technique for calibration of a classification model, and various scores evaluating the quality of the found hyperplane in each iteration will be presented.

#31: **Parallel inverse solver for ultrasound breast tomography**

Presenter: Vaclav Hapla
Co-authors: Naiara Korta Martiartu, Christian Boehm

Measurements of mechanical waves travelling through a medium can be used to reveal the subsurface and interior structure of unknown objects. This has plentiful applications ranging from medical imaging at millimetre scale to seismic tomography at the planetary scale. However, solving these problems is challenging from both a mathematical and computational perspective, and scalable simulation tools are key to enable scientific progress. We present an inverse solver for image reconstruction in Ultrasound Computed Tomography (USCT) for early breast cancer detection. USCT is a non-invasive, radiation-free, pressure-free and low-cost technique that uses both transmitted and reflected signals to create images of the soft tissue's acoustic properties. These images are particularly useful for characterizing interior breast tissue and differentiating between benign and malign lesions. A short time-to-solution, from taking measurements to obtaining the image, is crucial for any medical imaging technique. It must be in the order of minutes to be applicable in practice. In addition, the computational resources in a hospital are limited and should not exceed a dedicated workstation. To meet these requirements, we employ a simplified physical model using ray-tracing and apply time-of-flight tomography to reconstruct the acoustic properties of the breast tissue. This approach leads to a linear least-square problem with a large sparse rectangular matrix. The problem is in general ill-posed, which can be handled by various regularization strategies.
We were originally using MATLAB to assemble, regularize and solve this problem. However, this is undesirable in practice due to its strict and expensive licensing, and constraints on the parallel solution and computer architecture. To overcome this, we decided to use Portable, Extensible Toolkit for Scientific Computation (PETSc). It provides all needed ingredients: distributed vectors and sparse matrices, fast parallel assembly and linear algebra routines, and implementations of least-squares methods. PETSc has a permissive open source license (FreeBSD), and is highly portable - it can be used with virtually any relevant computer architecture, operating system, and toolchain.

CP 1.4.C CAB G 51 SPLITTING METHODS Chair: W. Gansterer

#32: **The generalized HSS method with a flexible shift-parameter for non-Hermitian positive definite linear systems**

Presenter: Guoyan Meng
Co-authors: Ruiping Wen

Based on the Hermitian and skew-Hermitian splitting (HSS), we come up with a generalized HSS iteration method with a flexible shift-parameter for solving the non-Hermitian positive definite system of linear equations. This iteration method utilizes the optimization technique to obtain the optimal value of the flexible shift- parameter at iteration process. Both theory and experiment have shown that the new strategy is efficient.

#33: **Adaptive resolution of linear systems based on a posteriori error estimators**

Presenter: Zakariae Jorti
Co-authors: Ani Anciaux-Sedrakian, Laura Grigori, Jan Papež, Soleiman Yousef

In many scientific applications, the resolution of partial differential equations usually leads to solving large sparse linear systems. This typically involves use of iterative or hybrid methods, which forcibly generate errors due to numerical approximations. These errors can have a non-uniform distribution in space and there might be some discrepancy in their magnitudes. In order to reduce the number of iterations and efficiently use the computational resources, one way could be to adaptively exploit the information about the distribution of the errors within the algebraic iterations. Since these errors are generally unknown, we suggest to rely on a posteriori error estimates instead, in particular, on the estimates that are computable locally on each mesh elements and can be decomposed in components that identify the source of the error e.g. discretization error, algebraic error. The distinction between those components allows derivation of adaptive algorithms, which ensure a reduction of calculations using proper stopping criteria for algebraic iterations, a balancing spatial distribution of errors and an adaptive mesh refinement. Several works done on equilibrated fluxes estimators go in that direction. In the present work, we propose an adaptive procedure adjusting to the problem thanks to the information on the algebraic error distribution. More specifically, we propose to target the regions where the algebraic errors were large and apply a technique to reduce them and converge faster than a standard solve.

Given an approximate solution and assuming we can tightly estimate the local distribution of the algebraic error on each mesh element, we decompose the main domain into two disjoint subdomains Omega1 and Omega2, such that the algebraic error in Omega1 is much greater than its counterpart in Omega2. We first algebraically formulate this starting hypothesis with the use of local stiffness matrices associated to subdomains Omega1 and Omega2. Next, we construct a 2x2 block splitting of the matrix, based on which we derive a modification of the starting hypothesis that involves SPD submatrices. This second formulation motivates the use of a Schur complement procedure for a targeted decrease of the dominant part of the error. The procedure is based on an exact factorization on one diagonal block of the matrix, coupled with a Schur complement of the remaining block. The resulting algorithm can be seen as a hybrid solver since it combines a direct solve on a subdomain and an iterative solve on its complementary. Then, we present an equivalent preconditioner that, combined with a PCG for the global system, is equivalent to the Schur procedure with PCG. We proceed with a cost analysis of the procedure in terms of number of arithmetic operations and compare it to a standard PCG solve. Some experimental tests done on 2D elliptic problems are presented to show that with the proposed procedure significant gains can be achieved. In the tests, we start by a couple of academic Poisson problems and then move on to more complex diffusion problems with inhomogeneous coefficients. We observe that the gain strongly depends on whether or not the error region was captured fully or partially in the subdomain on which exact factorization is performed. We conclude that this procedure seems to be most advantageous for problems where errors are not widely spread with contiguous concentration of high errors.

#34: **Parallel Multisplitting Iteration Methods Based on Optimization for Linear Systems**

Presenter: . . . . . Rui-Ping Wen
Co-authors: . . . . .

In this report, we not only want to decreases the difficulty of constricting the multisplitting of the coefficient matrix, but also releases the constrains to the weighting matrices. We present two optimization models to modify parallel multisplitting iteration methods for solving positive definite (symmetric or non-symmetric) linear systems.

#35: **Modulus-Based Parallel Multiplitting Iteration Methods for Linear Complementarity Problems**

Presenter: . . . . . Zhong-Zhi Bai
Co-authors: . . . . .

In order to solve large sparse linear complementarity problems on parallel multiprocessor systems, by making use of the modulus reformulation of the target problems and the multiple splittings of the system matrices we design the parallel modulus-based matrix splitting iteration methods, the modulus-based matrix splitting two-stage iteration methods and their relaxed variants. We prove the asymptotic convergence of these matrix multisplitting iteration methods for the H-matrices of positive diagonal entries, and give numerical results to show the feasibility and effectiveness of the modulus-based matrix multisplitting iteration methods when they are implemented in the parallel computational environments.

Thursday, 28.06.2018 10:45 AM - 12:25 PM Parallel Session 2.2.

MS 2.2.A CAB G 11 PARALLEL EIGENVALUE SOLVERS FOR LARGE SCALE PROBLEMS I Chair: P. Arbenz

#36: **The ChASE library for large Hermitian eigenvalue problems**

Presenter: Edoardo Di Napoli
Co-authors: Jan Winkelmann

In solving dense Hermitian eigenproblems arranged in a sequence, direct solvers fail to exploit the spectral properties of the problems which are the distinctive features of being part of a sequence. When such features take the form of correlations between the eigenvectors of consecutive problems, the potential benefit of exploiting them can be substantial. We present in this talk ChASE, a new library based on an optimized version of subspace iteration with polynomial acceleration. ChASE is a modern library written in C++ using the most current concepts in software engineering which favor a simple integration in application codes and effortless portability over heterogeneous platforms. When solving sequences of Hermitian eigenproblems for a portion of their exterior spectrum, ChASE experiences a considerable speedup and outperforms direct solvers in many scenarios. The library ships with two distinct parallelization schemes and is easily extensible to other computing architectures.

#37: **Solving large-scale eigenvalue problems in amorphous materials**

Presenter: Giuseppe Accaputo
Co-authors: Peter Arbenz, Peter Derlet

Amorphous solids, like metallic glasses, exhibit an excess of states in the low frequency regime very close to the boson peak, and the precise nature of these low frequency vibrations remains unclear.
In this paper we investigate the use of a polynomial filtered eigensolver for the computation and study of low frequency eigenmodes of a Hessian matrix located in a specific interval close to the boson peak regime. A distributed-memory parallel implementation of a polynomial filtered eigensolver is presented. Our implementation, based on the Trilinos framework, is then applied to Hessian matrices of different atomistic bulk metallic glass structures derived from molecular dynamics simulations for the computation of eigenmodes close to the boson peak. In addition, we demonstrate the parallel scalability of our implementation on multicore nodes. Our resulting calculations successfully concur with previous results, and anomalous behavior of the particles in the region close to the boson peak can be observed from the data.

#38: **The EVSL package for symmetric eigenvalue problems**

Presenter: Yousef Saad
Co-authors: Ruipeng Li, Yuanzhe Xi, Lucas Erlandson

A number of applications require the computation of tens of thousands of eigenvalues of very large matrices. For these problems, it is imperative to take advantage of spectrum slicing strategies whereby different 'slices' of the spectrum are extracted independently. The presentation will begin by describing a general approach for spectrum slicing based on polynomial filtering. This approach can be quite efficient in the situation where the matrix-vector product operation is inexpensive and when a large number of eigenvalues is sought. Polynomial filtering can be combined with the Lanczos algorithm with and without restarts, as well as with subspace iteration. An alternative to polynomial filtering that is generating a growing interest is a class of methods that exploit filtering by rational functions. Good representatives of this general approach are the FEAST eigensolver and the Sakurai-Sugiura algorithm. Here we will argue that the standard Cauchy integral–based approach can be substantially improved upon – especially when iterative solvers are involved. These two classes of techniques have recently been implemented in a code named EVSL (for eigenvalues slicing library) and the presentation will end with the latest updates to the code and our progress with its (forthcoming) parallel version.

#39: **Symmetry-preserving of the Hankel-type Sakurai-Sugiura eigenvalue solver for large sparse Hermitian definite generalized eigenvalue problem**

Presenter: Yasunori Futamura
Co-authors: Akira Imakura, Tetsuya Sakurai

In recent years, a contour integral-based method such as the Sakurai-Sugiura method attracts an attention due to its inherent distributed parallelism. A block Hankel-type variants of the Sakurai-Sugiura method (block SS-H) is one of the most inexpensive variant because of the absence of large sparse matrix vector multiplications and orthogonalizations at the projection phase. However, the small projected problem is not always a Hermitian definite problem even if the original generalized eigenvalue problem is Hermitian definite. In this presentation, we show a new block Hankel-type variant of the Sakurai-Sugiura method that can preserve the symmetry of the original problem. We also present a technique for improving accuracies of eigenvectors and show numerical examples using problems from practical applications.

MS 2.2.B CAB G 61 PARALLELIZATION ASPECTS OF SVD AND EVD COMPUTATIONS I Chair: M. Vajteršic

#40: **Parallel solution of the generalized eigenvalue problem given in a factored form**

Presenter: Sanja Singer
Co-authors: Edoardo di Napoli, Vedran Novaković and Gayatri Čaklović

The standard LAPACK algorithm for the generalized Hermitian eigenproblem

$$Ax = \lambda Bx,$$

where both matrices are Hermitian, and B is positive definite, performs a reduction to the standard eigenvalue problem by using the Cholesky factorization of B. Such an approach is not well suited for the parallel computation since the Cholesky factorization is inherently sequential.
In the FLAPW Method (Full Potential Linearized Augmented Plane Wave Method) – an electronic structure method in solid state physics – the obtained generalized eigenvalue problem is of the form

$$A = \sum_{k=1}^{n} A_k^{ast} T_k A_k, \quad B = \sum_{k=1}^{n} B_k^{ast} B_k,$$

where T_k are Hermitian and indefinite, and the matrices A_k^{ast} and B_k^{ast} are tall and skinny. Instead of explicitly forming A and B, and then computing the generalized eigendecomposition, another approach is to transform the problem into an implicit generalized eigenvalue problem, i.e., into a generalized SVD, of the following form

$$(A,B) := (F^{ast} JF, G^{ast} G),$$

with F and G that have approximately three times more rows than columns, and $J = mdiag(pm1)$.
We show how to modify the Hari–Zimmermann method for the generalized eigenproblem to work in parallel on A and B implicitly, i.e., not directly on the matrices A and B, but on their factors F, G, and J instead. The parallelization approach has three stages. First, the matrices T_k are factored in parallel by the Hermitian indefinite factorizations, revealing the sign matrix J. Then, the obtained factors are used to form F. Finally, the generalized SVD of a pair $(F^{ast} JF, G^{ast} G)$ is computed. As a preprocessing part of the last stage, the pair can optionally be "shortened" to the one with square matrices by the (indefinite) QR factorizations.

#41: **Asymptotic Quadratic Convergence of the Two-Sided Parallel Block-Jacobi SVD Algorithm**

Presenter: Gabriel Okša
Co-authors: Yusaku Yamamoto, Martin Bečka, Marián Vajteršic

We present the proof of the global and asymptotic quadratic convergence of the parallel two-sided block-Jacobi SVD algorithm with dynamic ordering.
When dealing with the convergence analysis of any SVD algorithm, it is sufficient to consider only square matrices. If an original matrix is of size $m \times n, m \geq n$, one can compute first its QR decomposition and then apply the iterative SVD algorithm to the square factor R of size n. The SVD of an original matrix can be then re-constructed in an obvious way.
Let us divide a square matrix A of order n into a $w \times w$ block structure with w blocks in each block row (column). Denote by A_{IJ} the (I,J)th block of size $ell \times ell, ell = n/w$. Hence, there are $w(w-1)$ off-diagonal blocks in A.
In the parallel case, having p processors, the blocking factor $w = 2p$ is chosen, so that each processor contains two block columns of matrix A. Using the greedy implementation of dynamic ordering, p pairs of the off-diagonal blocks with largest weights and disjunct block row and column indices are zeroed in each parallel iteration step by a pair of appropriate orthogonal transformations, which are applied in each 2×2 block sub-problem. We show that the off-diagonal Frobenius norm converges to zero quadratically in the asymptotic regime. Moreover, after defining the scaled iteration matrices, where a suitable diagonal scaling is applied from both sides of iteration matrices, we also show that the scaled off-diagonal Frobenius norm converges quadratically as well. Numerical examples confirm the developed theory.
Acknowledgment: This work has been supported by the VEGA Grant no. 2/0004/17.

#42: **A GPU variant of the implicit Hari–Zimmermann algorithm for the generalized SVD**

Presenter: Vedran Novaković
Co-authors:

The Hari–Zimmermann algorithm is a Jacobi-type method for computing the generalized eigenvalue decomposition (GEVD) of a matrix pair (A,B), where both matrices are Hermitian, and B is positive definite.
If A and B are given *implicitly* by their factors F, J, and G, respectively, such that $(A,B) = (F^{ast} JF, G^{ast} G)$, where the matrix $J = \mathrm{diag}(pm1)$ holds the signs of the eigenvalues of A on its diagonal, then the GEVD of (A,B) can also be computed implicitly, i.e., without assembling A and B in entirety from the factors, by a modification of the Hari–Zimmermann algorithm.
More precisely, the algorithm can be converted to a method that jointly orthogonalizes the pairs of columns of

F and G by a sequence of transformations that are applied only from one (right) side of the factors. Such a one-sided algorithm computes U, Σ_F, V, Σ_G, and Z, such that $FZ = U\Sigma_F$ and $GZ = V\Sigma_G$, where U is J-unitary ($U^{ast}JU = J$), V is unitary, and Σ_F and Σ_G are diagonal, real, and non-negative. In effect, the method implictly computes the GEVD of (A,B), but *explicitly* (when $J = I$) the generalized singular value decomposition (GSVD) of (F,G).
The recent work has shown that such method can be successfully blocked and parallelized for the CPUs with the shared memory, and for the clusters of those. Even the sequential blocked version outperformed the LAPACK's GSVD algorithm, and the parallel ones exhibited a decent scalability.
On the other hand, an efficient blocked variant of a one-sided Jacobi-type algorithm for the "ordinary" and hyperbolic SVD has been developed for the GPU(s), that utilizes the GPU almost fully, with the CPU serving only the controlling purpose.
This talk aims to merge the experience of those two approaches, and present a GPU-only parallel and blocked variant of the implicit Hari–Zimmermann algorithm for the GSVD as an extension of the latter.

#43: **Massively Parallel Polar Decomposition on Distributed-Memory Systems**

Presenter: ..... Hatem Ltaief
Co-authors: ..... Dalal Sukkari, Aniello Esposito, Yuji Nakatsukasa, and David Keyes

We present a high-performance implementation of the Polar Decomposition (PD) on distributed-memory systems. Building upon on the QR-based Dynamically Weighted Halley (QDWH) algorithm, the key idea lies in finding the best rational approximation for the scalar sign function, which also corresponds to the polar factor for symmetric matrices, to further accelerate the QDWH convergence. Based on the Zolotarev rational functions– introduced by Zolotarev (ZOLO) in 1877– this new PD algorithm ZOLO-PD converges within two iterations even for ill-conditioned matrices, instead of the original six iterations needed for QDWH. ZOLO-PD uses the property of Zolotarev functions that optimality is maintained when two functions are composed in an appropriate manner. The resulting ZOLO-PD has a convergence rate up to seventeen, in contrast to the cubic convergence rate for QDWH. This comes at the price of higher arithmetic costs and memory footprint. These extra floating-point operations can, however, be processed in an embarrassingly parallel fashion. We demonstrate performance using up to 102, 400 cores on two supercomputers. We demonstrate that, in the presence of a large number of processing units, ZOLO-PD is able to outperform QDWH by up to 2.3X speedup, especially in situations where QDWH runs out of work, for instance, in the strong scaling mode of operation.

MS 2.2.C CAB G 51 TASK-BASED PROGRAMMING FOR SCIENTIFIC COMPUTING I Chair: E. Agullo

#44: **Exploiting Nested Task-Based Parallelism in the Factorization of Hierarchical Matrices**

Presenter: ..... Enrique S. Quintana-Ortí
Co-authors: ..... Rocío Carratalá-Sáz, Sven Christophersen, Jose I. Aliaga, Vicenç Beltran, Steffen Börm

Hierarchical matrices (H-matrices) lie in-between the dense and sparse scenarios. Therefore, it is natural to tackle the LU factorization of H-Matrices via a task-parallel approach, which has reported successful results in the recent past for related linear algebra problems. Concretely, in this talk we leverage some recent features in the OmpSs-2 programming model, such as support for weak operands and early release of dependencies, to considerably improve the parallel efficiency when of H-LU factorizations arising from boundary element methods. Discovering the data-flow parallelism intrinsic to the operation at execution time, via the analysis of data dependencies based on the memory addresses of the tasks' operands, is especially challenging for H-matrices, as the data structures vary in dimension during the execution. We overcome this issue by decoupling the data structure from that used to detect dependencies.

#45: **High Performance Aynchronous Execution of the Reverse Time Migration for the Oil & Gas Industry**

Presenter: ..... Hatem Ltaief
Co-authors: ..... Issam Said, David Keyes

Task-based programming model is a promising alternative approach to remove artifactual synchronization points seen in bulk synchronous programming model. We would like to leverage the performance of the Reverse Time Migration (RTM) on GPU-based systems. By relying on a dynamic runtime system to schedule the various tasks of the RTM (e.g., stencil computation kernel, Perfectly Matched Layer computations, I/O operations, image condition calculations, etc.), the overall application translates into an out-of-order execution. This opens up new opportunities to further overlap expensive and non-critical operations, such as I/O, with tasks which belong to the critical path, such as high performance GPU stencil kernel computation during the forward/backward modeling. Idle time is then reduced, while load balancing is achieved through work stealing on each node. To further reduce the overhead of the I/O operations, numerical compression algorithms are investigated, in addition to the asynchronous execution, to prevent from running in an out-of-core mode of operation for maximum occupancy on GPU memory.

#46: **Limitations of OpenMP task-based parallelization to achieve high performance and create a robust software design**

Presenter: Bérenger Bramas
Co-authors:

OpenMP is likely to become the most used interface to develop task-based applications in the HPC and scientific computing communities. Using OpenMP provides multiple assets: it allows a low-dependency footprint, it is portable across compilers and, because it is standard, it relies on consistent and stable definitions. Consequently, its utilization provides important guarantees for long-term projects. On the other hand, the current OpenMP standard and its numerous implementations suffer from limitations to achieving high performance and to write clean and easy-to-maintain programs. These drawbacks come from the annotation system, the standard itself, or the lack of specific features. However, high performance is usually an objective that is intended to be achieved at all costs for many applications, and software design is increasingly less neglected as it is an asset that allows the creation of projects that will be used for several decades, involving researchers from different fields, and executed on constantly changing hardware. In this presentation, we will describe some of the limitations and performance weaknesses, and describe their impact on code development with the objective to highlight situations where this technology might not be the most appropriate. Among the targeted topics, we will focus on heterogeneity, scheduling, dependency management, and hardware abstraction. We will also point out possible solutions that exist in modern runtime systems to resolve most of these issues.

#47: **qr_mumps: a runtime based sparse direct solver for heterogeneous architectures**

Presenter: Alfredo Buttari
Co-authors: Emmanuel Agullo, Abdou Guermouche, Ian Masliah

qr_mumps is a parallel, direct solver for sparse linear systems based on the multifrontal QR factorization. Parallelism is achieved using a Sequential Task Flow (STF) programming model on top of the StarPU runtime system. In this talk we will show how STF parallelism can be applied to a sparse, direct solver and how the use of a modern runtime system allows for the portable and efficient implementation of complex algorithms that can improve its performance and scalability as well as its memory consumption. The effectiveness of this approach will be assessed through experimental results on multicore, manycore (Intel Knights Landing) and hybrid (multicore+GPU) platforms.

Thursday, 28.06.2018 01:45 PM - 03:45 PM Parallel Session 2.3.

MS 2.3.A CAB G 11 PARALLEL EIGENVALUE SOLVERS FOR LARGE SCALE PROBLEMS II Chair: E. Di Napoli

#48: **FEAST using residual inverse iterations with applications**

Presenter: Eric Polizzi
Co-authors:

The FEAST eigensolver uses complex contour integration and subspace iterations to calculate the eigenvectors whose eigenvalues that are located inside some user-defined region. The algorithm can be interpreted as a generalization of shift-and-invert iterations that uses multiple shifts in the complex plane leading to an optimal filter projector. In recent work (which includes the upcoming v4 of the software), the solver has been reimplemented to make use of residual inverse iterations. Although, the new filter form is mathematically equivalent to the original FEAST linear projector, it is numerically more efficient and more appealing in a number of new situations. We will demonstrate the effectiveness of the FEAST residual inverse iterations for addressing: (i) the inexact inner-outer iterative approach (IFEAST or FEAST without factorization), (ii) the mixed precision arithmetics iterative procedure, and (iii) the non-linear eigenvalue problem.

#49: **Block Krylov and Jacobi-Davidson methods on heterogenous systems**

Presenter: Jonas Thies
Co-authors: Melven Röhrig-Zöllner, Nigel Overmars, Dominik Ernst

Over the past five years we have developed two open source software packages called GHOST and PHIST (`https://bitbucket.org/essex/[ghost|phist]`). We discuss the software and performance engineering techniques used when designing these libraries and show some examples of use.
GHOST provides optimized implementations of memory-bounded linear algebra operations on heterogenous CPU/GPU systems. PHIST provides the software infrastructure for implementing iterative sparse matrix algorithms in a portable and efficient way by introducing a kernel interface layer inspired by the message passing interface (MPI). Implementations of the interface are verified using an extensive test suite and performance models. Going beyond the isolated optimzation of linear algebra kernels, phist allows algorithm-level performance optimizations like kernel fusion and overlapping of communication and computation.
To make phist algorithms easy to integrate into existing applications, we provide implementations of the kernel interface for various commonly used libraries such as Trilinos, PETSc and Eigen, and a Fortran+MPI reference implementation. Besides the standard C interface, Pyton, C++ and Fortran bindings are automatically generated for all functions.
We show how the new libraries can be used to boost the performance of existing implementations of Block Krylov solvers in the Trilinos package Anasazi, and present results for our own implementation of the block Jacobi-Davidson QR method applied to model problems from quantum physics.

#50: **A Golub-Kahan Davidson Method for Accurately Computing a Few Singular Triplets of Large Sparse Matrices**

Presenter: Andreas Stathopoulos
Co-authors: Eloy Romero

Obtaining high accuracy singular triplets for large sparse matrices is a significant challenge, especially when searching for the smallest triplets. Due to the difficulty and size of these problems, efficient methods must function iteratively, with preconditioners, and under strict memory constraints. In this research, we present a Golub-Kahan Davidson method (GKD), which satisfies these requirements and includes features such as soft-locking with orthogonality guarantees, an inner correction equation similar to Jacobi-Davidson, locally optimal +k restarting, and the ability to find real zero singular values in both square and rectangular matrices. Additionally, our method achieves full accuracy while avoiding the augmented matrix, which often converges slowly due to the difficulty of interior eigenvalue problems. We describe our method in detail, including implementation issues that may arise. Our experimental results confirm the efficiency and stability of our method over the current implementation of PHSVDS in the PRIMME software package.

#51: **Combining Refined and Harmonic Rayleigh-Ritz for Interior Hermitian Eigenvalue Problems**

Presenter: Eloy Romero Alcalde
Co-authors: Andreas Stathopoulos

Challenging large-scale interior eigenvalue problems appears in interesting physics models, e.g., in first-principle electronic structure analysis and quantum chromodynamics simulations. Conventional ways to tackle interior problems are based on rational filters, when the matrix problem factorization is affordable, and preconditioning methods (Generalized/Jacobi-Davidson and LOBPCG), for certain classes of eigenproblems that an efficient preconditioner is available. It is an open question how competitive are these methods without a factorization or a preconditioner.

This work summarizes our recent efforts on the methods used for computing the approximate eigenpairs in a subspace. These methods drives the convergence of preconditioning eigenmethods and have an important impact on the performance of Generalized Davidson and LOBPCG with poor preconditioners or unpreconditioned. We discuss the issues in terms of convergence and parallel performance presented in the standard approaches such as Harmonic and refined Rayleigh-Ritz together with the advanced refined harmonic approach introduced by Jia. Also we propose a heuristic to detect and address the problematic situations. The resulting method combines refined and harmonic, and as we show experimentally on large Hermitian problems, is more robust than the standard approaches and exhibit better parallel performance than Jia's methods, especially in combination with block methods.

MS 2.3.B CAB G 61 PARALLELIZATION ASPECTS OF SVD AND EVD COMPUTATIONS II Chair: G. Okša

#52: **New preconditioning for the parallel one-sided block-Jacobi SVD algorithm**

Presenter: Martin Bečka
Co-authors: Gabriel Okša

Parallel implementation of the one-sided block-Jacobi SVD algorithm (OSBJ) appeared to be an alternative to the `PDGESVD` procedure of `ScaLAPACK`. Recently we presented a new preconditioner for OSBJ in serial case. Since there are fast procedures for eigenvalue decompostion and matrix multiplication in `LAPACK/BLAS`, one can apply OSBJ to the matrix AV_1 instead of A, where V_1 is the orthonormal matrix of eigenvectors of A^TA. Columns of AV_1 should be more orthogonal than the columns of A. In finite arithmetic, level of their orthogonality depends on the 2-norm condition number $\kappa(A)$. Our serial preconditioned OSBJ was significantly faster than `DGESVD` as well as `DGESVJ` (Jacobi SVD from LAPACK) and also faster than `DGESDD` (divide and conquer SVD) in the case of small $\kappa(A)$. So a proper parallel implementation of this approach has a potential to overcome `PDGESVD`. We describe the approach in more detail and give experimental results.

#53: **Parallel reduction of four matrices to the condensed form for a general matrix eigenvalue algorithm**

Presenter: Nela Bosner
Co-authors:

The VZ algorithm proposed by Charles F. Van Loan (SIMA, 1975) is a QR-type process for the solution of the general matrix eigenvalue problem $ACx = \lambda BDx$, where $A,B \in \mathbb{R}^{n\times m}$, $C,D \in \mathbb{R}^{m\times n}$, and $m \geq n$. Especially, this algorithm is suitable for solving the generalized singular value problem $A^TAx = \mu^2 B^TBx$. Transforming the general eigenvalue problem to the standard form $(BD)^{-1}(AC)x = \lambda x$ represents a possible numerical danger, since formation of the products AC and BD, as well as formation of the inverse $(BD)^{-1}$ can produce a result with a large backward error. Thus, the VZ algorithm attempts to solve the problem without forming these products and inverse. This approach transforms each of the four matrices separately into a suitable form, which is a generalization of the Schur decomposition. Actually, the algorithm computes orthogonal matrices $Q,U \in \mathbb{R}^{n\times n}$ and $V,Z \in \mathbb{R}^{m\times m}$ such that QAZ is upper quasi-triangular, and QBV, Z^TCU and V^TDU are upper triangular. The VZ algorithm begins by reducing the matrices A, B, C, and D to an equivalent condensed form by the finite step initial reduction. This reduction finds orthogonal matrices Q_0, U_0, V_0 and Z_0, such that Q_0AZ_0 is upper Hessenberg, and Q_0BV_0, $Z_0^TCU_0$ and $V_0^TDU_0$ are upper triangular. Then, the VZ iterations are applied to the matrices in the condensed form. In the initial reduction, A is reduced to the upper Hessenberg form, while simultaneously preserving triangularity of the other three matrices. This is done by the Givens rotations, annihilating one by one element of A, and by generating three more rotations applied to the other matrices per each annihilation. Such an algorithm is quite inefficient. In our work, we propose a blocked algorithm for the initial reduction, based on the aggregated Givens rotations and matrix–matrix multiplications, which are applied in the outer loop updates. This algorithm has another level of blocking, exploited in the inner loop. Further, application of a sequence of the rotations in the inner loop is parallelized, with balanced operation count per thread. Since a large number of aggregated rotations is produced in every outer loop step, they are simultaneously accumulated before the outer loop updates. We also consider a variant of the algorithm in a hybrid CPU–GPU framework, where the compute-intensive outer loop updates are performed on GPU, and can be overlapped with the reduction in the next step performed on CPU. This adjustments speed up the original initial reduction considerably, and the efficiency of the whole VZ algorithm is increased.

#54: **A Parallel Generator of Non-Hermitian Matrices Computed from Given Spectra**

Presenter: Xinzhe Wu
Co-authors: Serge G. Petiton

Iterative or restarted linear algebra methods are the important parts of the global computing time of applications in various fields since decades, and the recent acceleration of researches associated with social networks, big data, machine learning and artificial intelligences increase the necessity of Non-Hermitian solvers associated with larger and larger sparse matrices. The convergence of the numerical linear system resolution and eigenvalue problem analysis of such problems is complex, and it is necessary to evaluate the convergence of iterative or

restarted methods to solve extremely large Non-Hermitian eigenvalue and linear problems on parallel and/or distributed machines. This convergence depends on the properties of spectra. Then, it is crucial to have large sets of matrices to test on these large-scale machines. These matrices should be generated with four special characteristics: 1) their spectra must be known and can be easily controlled; 2) they should be non-Hermitian and non-trivial; 3) they could have a very high dimension, including the non-zeros elements and/or the matrix size to evaluate on large-scale systems; 4) they should be sparse and the sparsity patterns must be controllable. In this talk, we present a method to generate sparse matrices from a given spectrum and matching some mathematical and shape properties. It is a scalable parallel matrix generator which uses the given spectra by users to build large-scale band matrices and to ensure their eigenvalues to be the given ones with high accuracy. The complexity of this method is $O(hdn)$ with n the dimension of matrix, h and d two parameters to define its bandwidth proprieties. The worst case would be an $O(n^3)$ problem for operations with large h and d, and it would require $O(n^3)$ memory storage. But if we want to generate a band matrix with h and d much smaller than the matrix dimension n, it turns to be an $O(n)$ problem with good potential scalability and to consume $O(n)$ memory storage. Since it is generated in parallel, the different parts of this matrix are already distributed on different computing units, it can be used directly to evaluate the parallel numerical method without concerning the I/O operations. This generator is firstly implemented both on CPUs and GPUs based on the PETSc, MPI, and CuSPARSE. Then an open source package SMG2S (Scalable Matrix Generator from Given Spectra) with specific communication optimized kernels based on MPI is also implemented. Strong and weak scaling performance of the different implementations is compared on top of the supercomputers TIANHE-2 in Guangzhou, China, and ROMEO in Reims, France. In order to verify its ability to keep the given spectra, we propose a check method based on the Shifted Inverse Power method. Good accuracy results are obtained by the verification experiments with the clustered spectrum, the closest eigenvalues, etc. Finally, we give an example which uses the SMG2S package to evaluate the GMRES method for solving non-Hermitian Linear Systems.

#55: **Spectrum slicing in quadratic symmetric eigenvalue problems**

Presenter: Jose E. Roman
Co-authors: Carmen Campos

In the linear symmetric-definite generalized eigenvalue problem it is possible to calculate all the eigenvalues contained in a given interval using the technique known as spectrum slicing citeCampos:2012:SSS. This technique is based on using the inertia concept to determine how many eigenvalues are located to the left of various selected points in the interval. At each point it is necessary to calculate a factorization and a Lanczos recurrence.
This scheme can be extended for quadratic eigenvalue problems with certain properties, for which the inertia can be computed [2]. In the case of definite quadratic eigenvalue problems (of which hyperbolic problems are a particular case), it is possible to form a symmetric-definite linearization $Ax = lambdaBx$, whose eigenvalues are all real [3]. Then, the spectrum slicing idea can be applied, provided that either A or B are positive definite. But getting a definite pair with this property may not be easy for large-scale problems [3]. An alternative is to adapt the spectrum slicing scheme to work with an indefinite pencil. In that case, the pseudo-Lanczos method must be used [4].
Preliminary results will be shown with an implementation in SLEPc [5]. This solver combines the pseudo-Lanczos iteration tailored for quadratic eigenproblems, with the computation of quadratic inertia in a spectrum slicing fashion.
Acknowledgements: Ministerio de Econom'ia y Competitividad under the project TIN2016-75985-P, as well as by European Commission FEDER funds.

[1] C. Campos and J. E. Roman. Strategies for spectrum slicing based on restarted Lanczos methods. *Numer. Algorithms*, 60(2):279–295, 2012.
[2] A. Kostić and H. Voss. On Sylvester's law of inertia for nonlinear eigenvalue problems. *Electron. Trans. Numer. Anal.*, 40:82–93, 2013.
[3] N. J. Higham, D. S. Mackey, and F. Tisseur. Definite matrix polynomials and their linearization by definite pencils. *SIAM J. Matrix Anal. Appl.*, 31(2):478–502, 2009.
[4] C. Campos and J. E. Roman. Restarted Q-Arnoldi-type methods exploiting symmetry in quadratic eigenvalue problems. *BIT*, 56(4):1213–1236, 2016.
[5] V. Hernandez, J. E. Roman, and V. Vidal. SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems. *ACM Trans. Math. Software*, 31(3):351–362, 2005.

MS 2.3.C CAB G 51 TASK-BASED PROGRAMMING FOR SCIENTIFIC COMPUTING II Chair: A. Buttari

#56: **Task-Based Sparse Direct solver for Symmetric Indefinite Systems**

Presenter: Florent Lopez
Co-authors: Iain Duff

Many applications in science and engineering require the solution of large sparse linear systems of equations. For solving such problems, direct methods are frequently employed because of their robustness, accuracy and usability as black-box solvers.

As modern architectures become more and more complex, with an increasing number of cores per chip, a deeper memory hierarchy and the integration of accelerators such as GPUs, it becomes all the more challenging to exploit the potential performance of such machines for sparse matrix factorization algorithms especially in the context of symmetric indefinite systems. Although significant efforts has gone into positive-definite systems, little progress has been reported in the much harder indefinite case. One major advance for tackling these problems is the design of the APTP (a posteriori threshold pivoting) strategy that has been implemented in the SSIDS solver and proven to be both efficient on multicore architectures compared to the state-of-the-art direct solvers.
In this talk, we present the DAG-based solver SpLDLT that relies on a APTP strategy and uses the StarPU runtime system for implementing it parallel version. We show the benefits of our approach for exploiting heterogeneity in the the context of GPU-accelerated multicore systems.

#57: **Design and analysis of a fully task-based application on modern HPC platforms: case study with a seismic kernel**

Presenter: Salli Moustafa
Co-authors: Wilfried Kirschenmann

Modern computing platforms are built from highly parallel processors and heterogeneous devices such as Graphics Processing Units and Many Integrated Cores. Moving forward to exascale platforms, estimations indicate that typical computing nodes will contain up to $\mathcal{O}(1000)$ cores. However, to fully take into advantage this large amount of computational power, a profound shift on the implementation of numerical applications has to be performed. Indeed, harnessing such systems usually implies mixing several programming paradigms, following the classical MPI+X approach. In this case, X can be OpenMP, Intel TBB, CUDA or OpenCL, and used for addressing a single computing node, whereas MPI is used for managing communications through the network interconnect. Task-based approach coupled with a generic runtime system is an emerging programming paradigm that greatly improves programmer productivity, leaving him to focus on the algorithm and computational kernels implementation. Following this trend, our goal in this work is to study the design and implementation of efficient scientific applications capable to scale at high core count and whose performances are portable across a large number of hardware architectures. From this perspective, we considered a task-based programming model coupled with PaRSEC, a generic task-based runtime system targeting distributed heterogeneous architectures. Such an approach allows to isolate the major concerns arising in scientific computing: the algorithm definition, the data distribution and the development of computational kernels. We present an end-to-end task-based algorithm for the seismic wave propagation including the time-step dependency within tasks definition. The algorithm data-flow contains seven types of tasks, characterized by different arithmetic intensities. Therefore, to improve the tasks scheduling, we theoretically evaluated the task priorities according to both its computational workload and its distance to tasks belonging to the critical path. To highlight the benefit of our design choices, we conducted an in-depth analysis of the impact of prioritizing tasks on the implementation performance. We present a strong scaling study on both shared and distributed memory platforms.

#58: **DPLASMA: dense linear algebra package for distributed heterogeneous systems**

Presenter: George Bosilca
Co-authors:

#59: **Towards Distributed Tasking in the PLASMA Numerical Library**

Presenter: Mawussi Zounon
Co-authors: George Bosilca, Jack Dongarra, Reazul Hoque

PLASMA (Parallel Linear Algebra Software for Multicore Architectures) is an established numerical linear algebra library for shared memory multicore and manycore architectures. Since its origins around 2007, PLASMA has been an early adopter of the task-based programming model. The recent tasking features of the OpenMP standard have been introduced in the latest version of the library.
Due to the limitation of OpenMP to shared memory architectures, we have recently explored different runtime systems with support for distributed tasking. In particular, we have ported PLASMA to the StarPU runtime as well as to the recent Dynamic Task Discovery interface of PaRSEC with a minimal intrusiveness to the shared memory version.
In this talk, we summarize our experience with using these two task-based programming libraries and discuss preliminary performance results.

Friday, 29.06.2018 08:45 AM - 10:45 AM Parallel Session 3.1.

CP 3.1.A CAB G 11 PARALLEL AND DOMAIN DECOMPOSITION LINEAR SYSTEM SOLVERS Chair: S. Margenov

#60: **Preconditioning Algebraic Optimized Schwarz Methods for Black-Scholes Partial differential Equations**

Presenter: Lahcen Laayouni
Co-authors:

In this paper we investigate the numerical solutions of the Black-Scholes partial differential equations. We are interested in solving the Option Pricing Model using the Algebraic Optimized Schwarz domain decomposition methods (AOSM). We will consider the European Vanilla Call and Put Options. Both semi and full implicit schemes in time will be considered. At each time step we will have to solve large-scale linear systems. The AOSM methods will be used as preconditioners to approximate the solutions of the obtained linear systems. The main idea of AOSM methods is based on replacing the classical transmission blocks by adequate blocks obtained from the neighbor sub-domains. The convergence of the optimal AOSM method in the case of two sub-domain decomposition is in two iterations for the present model. We will present also variants of the AOSM methods corresponding to different approximations of the transmission blocks. To accelerate the numerical computations we will the approximation of the transmission blocks will be computed using GPU computing. Numerical evidences show tremendous gain in timing when adopting the graphical computing.

#61: **Projector-avoiding TFETI for contact problems implemented in the PERMON toolbox**

Presenter: David Horak
Co-authors: Zdenek Dostal, Vaclav Hapla, Jakub Kruzik, Radim Sojka, Martin Cermak, Marek Pecha

The original FETI-1 (Finite Element Tearing and Interconnecting) method proposed by Farhat and Roux turned out to be a very powerful method for the parallel solution of problems described by elliptic PDEs. It is numerically scalable thanks to projectors onto the kernel of the natural coarse space. The authors proved later theoretically the bounds on the spectrum in terms of the ratio of the decomposition and discretization parameters.
The Total-FETI (TFETI) method developed by Dostal et al. uses Lagrange multipliers to enforce Dirichlet boundary conditions. This enables simpler assembly of the stiffness matrix kernel since for each subdomain it can be formed directly from the subdomain rigid body modes. The decomposition into a larger number of subdomains not only improves the bounds (and therefore reduces the number of solver iterations) but also reduces the time of stiffness matrices factorizations and their subsequent solve functions. However, the negative effect is an increase of the coarse problem (CP) size hidden in the projector application, so that the factorization of the CP matrix and subsequent solves become a bottleneck of the overall parallelization. Solving large CPs gets very complicated for tens or hundreds of thousands of subdomains, even if the best techniques are employed, e.g. parallel direct solvers (MUMPS, SuperLU_DIST) on sub-communicators or the HTFETI method, reducing the CP size by aggregating a small number of neighbouring subdomains into clusters.
Quadratic programming problems resulting from applying the TFETI method to variational inequalities can be solved by the MPRGP (Modified Proportioning with Reduced Gradient Projections) algorithm and SMALBE (Semi-Monotonic Augmented Lagrangian algorithm for Bound and Equality constraints) algorithm, both developed by Dostal. These algorithms have the rate of convergence given by the bounds on the spectrum of the Hessian matrix. This operator contains three projector applications that can be implemented using two CP solutions. In combination with TFETI, these algorithms were proved to enjoy both numerical and parallel scalability. These quadratic programming algorithms and FETI methods are implemented in our software package based on PETSc called PERMON (Parallel, Efficient, Robust, Modular, Object-oriented, Numerical) toolbox [`http://permon.vsb.cz`]. This presentation deals with the modification of the TFETI method eliminating projectors applications including CP solution while preserving the numerical scalability. Crucial for this achievement is a possibility to use the Moore-Penrose pseudoinverse of the stiffness matrix. This modification is obtainable through the projection of a generalized inverse onto the range of the stiffness matrix. This operation is purely local and very cheap. The CP solution contained in the penalized term of the Hessian of the quadratic programming problem ensuring the homogenized equality constraint satisfaction can be solved inexactly, which corresponds to the multiplication of homogenized equality constraints by some transformation matrix. An effect of various transformations of the equality constraints will be analysed. The performance of this new approach will be demonstrated by numerical experiments. Our tests of the projector-avoiding method show about 1.7 speedup over standard TFETI on problems with more than 1.25 billion of unknowns computed on up to 15,625 cores.

#62: **Parallel solution of sparse linear systems to find the shortest path in large scale graphs**

Presenter: Hilal Arslan
Co-authors:

Solving the shortest path problem on large scale networks is crucial for many applications. As parallelism became more common with the advent of multi-core architectures as well as large and complex networks have begun to

emerge in many settings, it is inevitable to come up with algorithms that take advantage of the current architectures. One alternative to solve the shortest path problem is to use one of the classical or improved parallel variations of the Dijkstra's algorithm. However, when the size of the network becomes large, finding the shortest path requires excessive computational time. Recently, some bio-inspired methods to find the shortest path have been proposed, such as Genetic algorithms, Ant Colony Optimization, Swarm Systems, and Physarum Solver.
Physarum Solver is capable of finding the shortest path in a labyrinth and is developed by modelling the behavior of Physarum Polycephalum, which is an amoeba-like organism. Physarum Solver has been applied in many applications recently. It can efficiently solve a variety of network optimization problems such as the traveling salesmen problem, the vehicle routing problem, and scheduling of multi-gravity assist trajectories and various optimization problems required linear programming. However, earlier studies provide only sequential variants of Physarum Solver.
In this study, a parallel and scalable Physarum Solver is proposed with the objective to find the shortest path for static graphs with positive edge weights. The proposed scheme is applied on both large scale realistic and real world static graphs as well as dynamically changing graphs. Physarum Solver requires the solution of the linear systems whose coefficient matrix is an M-matrix at each iteration. This step is the most time consuming step especially for problems having excessive data or information size. However, Physarum related studies in the literature do not take advantage of M-matrix property of the coefficient matrix to solve the linear systems in Physarum Solver. They use a direct method to solve such systems, which is infeasible for large scale problems with several millions of unknowns. A parallel preconditioned iterative method for solving prementioned sparse linear systems is presented. The proposed preconditioner is specifically designed based on the properties of the coefficient matrix of those linear systems, and the effectiveness of the proposed preconditioner is compared against other state-of-the-art preconditioners on dynamic graphs. Furthermore, the proposed dynamic algorithm is designed to be suitable for dynamically changing graphs since it uses the information arising in earlier iterations. The parallel scalability as well as the effect of changing the edge weights to the time to solution are evaluated for each graph model, separately and compared against a state-of-the-art parallel implementation of the Dijkstra's algorithm on a parallel multicore cluster. In contrast to the classical shortest path algorithms, the proposed scheme has a distinct advantage that it is using array based data-structures and optimized kernels which take advantage of today's multi level cache hierarchies. Our implementation exhibits remarkable speedups with comparable accuracy for synthetic and real-world applications.

#63: **Numerical Methods and Parallel Algorithms for Fractional Diffusion Problems**

Presenter: Svetozar Margenov
Co-authors:

We consider fractional powers of self-adjoint elliptic operators. The case of power $\alpha \in (0,1)$ is related to super-diffusion. In what follows we assume the definition based on spectral decomposition of the elliptic operator. The same approach is applied to define fractional powers of SPD matrices. In a general setting, the numerical solution of such nonlocal problems is rather expensive. The following four approaches lead to transformation of the equation $\mathcal{L}^\alpha u = f$ to some auxiliary local problem(s) in a computational domain of higher dimension: (A1) Extension to elliptic problem in a semi-infinite cylinder [2]; (A2) Transformation to a pseudo-parabolic problem [6]; (A3) Integral representation of the solution [1]; (A4) Best uniform rational approximation - BURA [4].
New efficient solvers for the linear system $\mathcal{A}^\alpha \mathbf{u} = \mathbf{f}$, $0 < \alpha < 1$, are proposed in [4], where $\mathcal{A}$ is a normalized symmetric and positive definite (SPD) matrix generated by finite element or finite difference approximation of the self-adjoint elliptic operator $\mathcal{L}$. Instead of the original problem, the system $\mathcal{A}^{\alpha-\beta}\mathbf{u} = \mathcal{A}^{-\beta}\mathbf{f} := \mathbf{F}$, $\beta \geq 1$ - integer, is considered. Then $\mathcal{A}^{\beta-\alpha}\mathbf{F}$ is approximated by a set of solutions of systems with $\mathcal{A} + d_j I$, $d_j \geq 0$, $j = 1, \ldots, k$, where $k \geq 1$ is the number of partial fractions of BURA $r_\alpha^\beta(t)$ of $t^{\beta-\alpha}$, $t \in (0,1]$. From algorithmic point of view, the methods (A3-A4) are very similar. This holds true for their parallel implementation as well. Comparing the accuracy, some advantages of the BURA methods are observed. This is stronger expressed for stronger super-diffusion.
The parallel implementation makes the fractional diffusion models more feasible and practically applicable. The parallel efficiency requires appropriate algorithms. The selection of best fitted solutions needs extended scalability analysis on the targeted parallel architectures. First scalability study of parallel algorithms for methods (A2-A3) is presented in [3]. Not surprisingly, better strong scalability is reported for the second method. In this case, the algorithm is based on a two-level parallelization template. At the first level, a number of independent local (sparse) subproblems are solved, while at the second level, parallel multigrid solvers are employed for each subproblem. Parallel scalability results for BURA are presented in [4]. A more involved performance analysis on Intel Xeon Phi towards scalability for extreme scale problems is provided in [5]. The last part of the talk is devoted to analysis of parallel efficiency accounting for the targeted accuracy.
The partial support by Grant No. BNSF-DN12/1 is acknowledged.

[1] A. Bonito, J. Pasciak, *Numerical Approximation of Fractional Powers of Elliptic Operators*, Mathematics of Computation, 84 (2015), 2083-2110

[2] L. Chen, R. Nochetto, O. Enrique, A.J. Salgado, *Multilevel Methods for Nonuniformly Elliptic Operators and Fractional Diffusion*, Mathematics of Computation, 85 (2016), 2583-2607

[3] R. Ciegis, V. Starikovicius, S. Margenov, R. Kriauziene, *Parallel Solvers for Fractional Power Diffusion Problems*, Concurrency Computat: Pract Exper (2017), https://doi.org/10.1002/cpe.4216

[4] S. Harizanov, R. Lazarov, S. Margenov, P. Marinov, Y. Vutov, *Optimal Solvers for Linear Systems with Fractional Powers of Sparse SPD Matrices*, Numerical Linear Algebra with Applications (2018), DOI: 10.1002/nla.2167

[5] N. Kosturski, S. Margenov, Y. Vutov, *Performance Analysis of MG Preconditioning on Intel Xeon Phi: Towards Scalability for Extreme Scale Problems with Fractional Laplacians*, Large-Scale Scientific Computing, Springer LNCS, 10665 (2018), 304-312

[6] P.N. Vabishchevich, *Numerically Solving an Equation for Fractional Powers of Elliptic Operators*, Journal of Computational Physics, 282 (2015), 289-302

CP 3.1.B CAB G 61 KRYLOV SPACE METHODS Chair: D. Gordon

#64: **Parallel performance and numerical stability of communication-hiding Krylov subspace methods**

Presenter: Siegfried Cools
Co-authors: Jeffrey Cornelis, Pieter Ghysels, Wim Vanroose

Krylov subspace methods are frequently used as efficient iterative solution methods for large scale linear systems occurring in a variety of HPC applications. A clear trend in current (petascale) and future (exascale) HPC hardware is the continuous up-scaling of the number of parallel compute nodes. The performance of Krylov subspace methods on these massively parallel systems is hence limited by global synchronizations and communication latency (stemming from the calculation of dot-products in the algorithm) rather than the floating point performance for which they were primarily optimized in the past.
Several possible communication reducing alternatives to classic Krylov subspace methods have (re-)gained significant attention over the past years. These include the class of s-step Krylov subspace methods [1] that aim to reduce the number of global synchronization bottlenecks in the iteration. In addition to "avoiding" communication, another approach to improve parallel scalability is to overlap time-consuming global reduction phases with useful (local) computations, thus reducing the impact of communication latency and decreasing the total time to solution. The latter technique has been implemented in the so-called pipelined Krylov subspace methods [2–4], in which global communication latency is "hidden" behind SPMVs and local vector operations.
Although it has been shown that these re-engineered communication-reducing Krylov subspace methods indeed offer improved parallel scalability [2,3,5], this performance gain often goes hand in hand with reduced numerical stability. Pipelined Krylov subspace methods, for example, use multi-term recurrence relations to compute the auxiliary variables required to construct the 'pipeline'; and overlap global communication. Theoretically, i.e. in exact arithmetic, pipelined methods produce a series of iterates that is identical to the traditional Krylov subspace iterates. However, in a practical finite precision setting the propagation of local rounding errors may affect numerical stability significantly.
In this talk we give an overview of the design, performance and numerical properties of communication-hiding pipelined Krylov subspace methods. We focus specifically on pipelined CG methods [3,4] to illustrate our approach. A numerical stability analysis explains the loss of maximal attainable accuracy that is observed in pipelined Krylov subspace methods [6,7]. Based on this analysis possible countermeasures to increase numerical stability – while aiming to retain the improved the parallel scalability obtained by pipelining – are suggested. Numerical accuracy and parallel performance experiments demonstrate the practical use of the analytical results.

[1] E. Carson, N. Knight, and J. Demmel. Avoiding communication in nonsymmetric Lanczos-based Krylov subspace methods. SIAM J. Sci. Comput., 35(5):S42–S61, 2013.
[2] P. Ghysels, T.J. Ashby, K. Meerbergen, and W. Vanroose. Hiding global communication latency in the GMRES algorithm on massively parallel machines. SIAM J. Sci. Comput., 35(1):C48–C71, 2013.
[3] P. Ghysels and W. Vanroose. Hiding global synchronization latency in the preconditioned Conjugate Gradient algorithm. Parallel Computing, 40(7):224–238, 2014.
[4] J. Cornelis, S. Cools, W. Vanroose. The communication-hiding Conjugate Gradient method with deep pipelines. SIAM J. Sci. Comput. (submitted). Preprint available at: `https://arxiv.org/abs/1801.04728`.
[5] P.R. Eller and W. Gropp. Scalable non-blocking preconditioned Conjugate Gradient methods. In SC16: Int. Conf. for HPC, Networking, Storage and Analysis, pages 204–215. IEEE, 2016
[6] 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 Gradient method. SIAM J. Mat. Anal. Appl., 39(1):426–450, 2018.
[7] S. Cools. Numerical stability analysis of the class of communication hiding pipelined Conjugate Gradient methods. SIAM J. Sci. Comput. (submitted). Preprint available at: `https://arxiv.org/abs/1804.02962`.

#65: **Fast estimation of statistical leverage scores by block iterative methods and randomization**

Presenter: Aleksandros Sobczyk
Co-authors: Efstratios Gallopoulos

We describe recent algorithms for estimating the statistical leverage scores and coherence of full rank matrices. These measures are of importance in large scale data mining, graph analytics and machine learning applications. The algorithms use the orthogonal projector "hat matrix"; and are based upon an efficient preconditioned block conjugate gradient solver, supercharged by state-of-the-art stochastic techniques from randomized numerical linear algebra. We use these techniques to build effective preconditioners, to enable the reduction of the number of right-hand sides in the block method, and to prove probabilistic upper bounds for the error of the computed

solutions. The algorithms are well suited for tall and thin sparse matrices and their dominant kernel is matrix multiplication. We present implementations and evaluate their performance on parallel and distributed platforms using real world and synthetic datasets and show that they are competitive with other state-of-the-art methods.

#66: **Application of deflated Newton-Krylov methods to the problem of finding bifurcation points**

Presenter: Michiel Wouters
Co-authors: Wim Vanroose

In numerical continuation of the steady states of large scale dynamical systems one is often interested in finding bifurcation points. These are points along the curve where the Jacobian of the equation is singular. Classic Newton methods like Newton-Krylov, the method of choice under normal circumstances, are strongly hampered by the singularity of the Jacobian at the bifurcation point. Certainly in cases where multiple eigenvectors with zero eigenvalues exists in this point. Adapting these methods with a deflated decomposition or a line-search may improve the convergence properties in some cases. However, in practice the resulting algorithms are often still too slow or do not converge up to a desired tolerance.
In the current talk further adaptations and refinements of the deflated Newton-Krylov method are investigated, based on splitting the update vector into a part that lies in the Jacobians range, and parts that correspond to approximate zero eigenvectors. If the line-search algorithm is only applied to certain parts of the vector, convergence often improves. In order to effectively apply the techniques on the bifurcation problem, it is required to provide them in combination with block elimination routines as well, which will be illustrated in the presentation. Examples from pattern formation in superconductors are used as illustrations of the methods.

#67: **Some recent results on accelerated parallel projection methods**

Presenter: Dan Gordon
Co-authors: Rachel Gordon, The Technion–I.I.T.

We present some inter-related results on accelerated parallel projection methods. The first result is called the Cimmino-Kaczmarz Equivalence, meaning that the Cimmino algorithm, which consists of projections and vector averaging, is equivalent to the Kaczmarz algorithm (which uses only projections) in some superspace of the problem space. The practical consequence of this is that in the CARP-CG algorithm [G & G, PARCO 2010], the internal Kaczmarz processing of the subdomains can be replaced by Cimmino, which is more amenable to fast computing on a GPU. The Kaczmarz-Cimmino equivalence implies a formal convergence proof for such a modification. This result is quite general, allowing different relaxation parameters to be used in the Cimmino algorithm.
A second result concerns the CGMN algorithm [Björck & Elfving, BIT 1979] and its block-parallel version CARP-CG. Both are examples of the Kaczmarz algorithm accelerated by CG, as follows: by running Kaczmarz in a forward and backward sweep, the resulting iteration matrix is symmetric and positive semi-definite, so the process can be accelerated by CG. Kaczmarz is, in fact SOR ran on AA^T, where A is the normalized system matrix. Although the use of AA^T is generally not recommended (because its condition number is the square of the condition number of A), it has been shown in previous work that after A is normalized, the diagonal elements of AA^T are all 1, and the off-diagonal elements are ¡ 1. Our new result uses two model problems – a convection-dominated elliptic PDE and a high-frequency Helmholtz equation – to demonstrate that when a certain problematic parameter increases, the maximal off-diagonal element of a row of A increases unboundedly w.r.t. the diagonal, while in AA^T, the maximal off-diagonal element remains well-bounded below the diagonal . The ”problematic parameter” is the size of the convection term in the elliptic PDE, and the frequency in the Helmholtz equation.
A third result concerns a major issue in domain decomposition (DD): the problem of eliminating inaccuracies caused by integrating the subdomain solutions across subdomain boundaries. An even harder problem arises in the case of cross points, at which three or more subdomains meet. This topic has received a lot of attention in recent years, with several problem-specific solutions. It is shown, and formally proved, that these problems do not exist with the CARP-CG algorithm. This is due to the fact that in CARP-CG, both the local processing and the merging of the local solutions are actually solved in a certain superspace in a unified manner. Furthermore, there is no need for any problem-specific adaptation. The concept of component-averaged DD (CADD) generalizes CARP-CG by allowing the use of other methods – besides Kaczmarz or Cimmino – for the internal processing in the subdomains. Sufficient conditions for the convergence of a CADD method are discussed.

CP 3.1.C CAB G 51 MIXED PRECISION AND LIBRARIES Chair: A. Basermann

#68: **AVX2 acceleration of SpMV and vector operations with Double-double precision vectors**

Presenter: Hidehiko Hasegawa
Co-authors:

High precision arithmetic operations reduce rounding errors and may improve the convergence of iterative methods, however high precision arithmetic operations are difficult to implement and costly. Double-Double (DD) precision arithmetic, one of high precision arithmetic, is easy to implement but still costly.
We developed a library, called “DD-AVX”, which includes DD precision vector and double precision sparse matrix operations accelerated by SIMD AVX2. In many situations, a coefficient matrix A is given in double precision

and used without modification during iterative process. Restricting a coefficient matrix in double precision, we can reduce memory consumption and the number of arithmetic operations. The number of bytes per flop is also improved.
As results, the computation time of multiplication of a double precision matrix and DD vector becomes about 2/3 times that of DD matrix and DD vector, and about 1.3 times that of both in double precision. The ratio may depend on the structure of sparse matrices.
In this talk, we introduce the interface of "DD-AVX" and a performance result of mixed precision arithmetic operations.

#69: **The Sparse Matrix Multiplication DBCSR library**

Presenter: Alfio Lazzaro
Co-authors: Jürg Hutter, Ilia Sivkov

Multiplication of two sparse matrices is a key operation in the simulation of the electronic structure of systems containing thousands of atoms and electrons. The highly optimized sparse linear algebra library DBCSR (Distributed Block Compressed Sparse Row) has been specifically designed to efficiently perform such sparse matrix-matrix multiplications. The library is designed to efficiently perform block-sparse matrix-matrix multiplication of matrices with a relatively large occupation. This library is the basic building block for linear scaling electronic structure theory and low scaling correlated methods in CP2K framework. It is parallelized using MPI and OpenMP, and can exploit GPU accelerators by means of CUDA. It is written in Fortran and is freely available under GPL license at github.com repository. Here we introduce the library implementation and parallelization strategies. In particular, we will describe a communication-reducing algorithm for multiplications involving rectangular and square matrices. We also present performance results where the tests are performed within the CP2K package with application benchmarks. These tests imply multiplication of square and rectangular sparse matrices with different sparsity values and matrices sizes. Finally, we will compare the performance with other scientific packages for multiplication of dense and sparse matrices.

#70: **Alien: a Flexible Wrapper API on Linear Solvers**

Presenter: Cédric Chevalier
Co-authors: Sylvain Desroziers, Jean-Marc Gratien, Pascal Havé, Xavier Tunc

We present our work on software engineering for numerical simulations codes and linear solvers libraries. By taking the same path that has conducted to design the C++ framework Arcane for our simulation codes, we ended up to create a new API, Alien, to interact with linear solvers libraries. We will describe its key design concepts and will then discuss how it can be extended to also become a framework to benchmark linear algorithms and libraries.
Writing *large* parallel distributed codes to compute complex physical phenomena is notariously a difficult task. However, several approaches have emerged to improve developer productivity, essentially by using abstract description of the parallelism wished by the programmer and then by generating all the supporting layer. For example, our application developers use a C++ framework named Arcane that provides basic utilities (Array, String, Mesh), application management (Time loop, IO) and parallel abstraction (Parallel Manager). When using distributed memory programming, Mesh data structure is naturally distributed accross MPI processes and code developer does not have to call any MPI function: high level functionalities provided by Arcane are sufficient... Except when dealing with linear Algebra.
Application developer was on his own to interact with linear solver libraries. He has to:

- deal directly with data distribution and MPI calls;
- implement specific data structures for each library;
- identify what functionalities are available in each library.

Furthermore, we had to satisfy more functionalities:

- user wants to be able to switch between linear solvers and algorithms, if possible at run time;
- application code must be independent of a specific linear solver;
- complex assemblies that occur in tightly coupled multi-physics codes must be handled easily and efficiently.

Whereas PETSc or Trilinos already provide access to a wide range of solvers and libraries, their approaches are still mostly "solver oriented", and deeply linked with the implementations.
Our answer is Alien, a C++ wrapper over linear solver libraries: we do not implement any solver, we just provide access to external solvers. Its key design concept is the notion of multi representations. For example, a Matrix is an object that can have several representations/implementations we can choose dynamically.
We will present and justify our design choices:

- we use simple objects for each functionality (in the spirit of UNIX KISS philosophy);
- functional extension is done by adding new objects;
- how we ensure that objects are used in a coherent way;
- API is designed to be asynchronous.

We will also discuss how Alien has allowed us to add high level functionalities, to all external solvers, such as data redistribution, to be able to exploit only specific computing resources for linear computations.
This made us realize that even if Alien was primarily designed for numerical simulation developers, it might also be used in other ways. For example, the linear solver community can see it as a mean to share an advanced bench-

marking framework. Several solvers can be accessed from a common API and we can easily switch between algorithms and libraries to compare. Currently, we have already plugged (part of) Hypre, PETSc, Trilinos, MTL4, but we are willing to continue with other solvers.

#71: **Mixed-Precision In-Memory Computing**

Presenter: . . . . . Costas Bekas
Co-authors: . . . . . Manuel Le Gallo, Abu Sebastian, Roland Mathis, Matteo Manica, Heiner Giefers, Tomas Tuma, Alessandro Curioni, Evangelos Eleftheriou

As the CMOS scaling laws break down because of technological limits, a radical departure from the processor-memory dichotomy is needed to circumvent the limitations of today's computers. In-memory computing is a promising concept in which the physical attributes and state dynamics of nanoscale resistive memory devices organized in a computational memory unit are exploited to perform computational tasks with collocated memory and processing. However, device variability and non-ideal device characteristics pose technical challenges to reach the numerical accuracy usually required in practice for data analytics and scientific computing. To resolve this, we propose the concept of mixed-precision in-memory computing that combines a von Neumann machine with a computational memory unit in a hybrid system that benefits from both the high precision of digital computing and the energy/areal efficiency of in-memory computing. We demonstrate the efficacy of this approach by addressing the problem of solving systems of linear equations and present experimental results of solving accurately a system of 5,000 equations using 998,752 phase-change memory devices. Our studies illustrate that a judicious interconnection of high-precision arithmetic and in-memory computing can be used to solve problems at the core of today's computing applications.

Friday, 29.06.2018 11:15 AM - 01:15 PM Parallel Session 3.2.

MS 3.2.A CAB G 11 RECENT ADVANCES IN PARALLEL SPARSE DIRECT SOLVERS Chair: E. Ng

#72: **symPACK: A new parallel sparse symmetric linear solver**

Presenter: Esmond G. Ng
Co-authors: Mathias Jacquelin

In this talk, we will describe a new parallel solver, symPACK, for solving sparse symmetric linear systems using Cholesky factorization on distributed-memory platforms. It differs from most of the existing solvers in that our implementation is task based, and uses 1-sided communication and dynamic scheduling. We will discuss the design of symPACK and provide preliminary results to demonstrate its performance.

#73: **Efficient Parallel Implementation of Spectral Nested Dissection for Large-Scale Sparse Linear System**

Presenter: Yuta Inagawa
Co-authors: Yasunori Futamura, Akira Imakura, Tetsuya Sakurai

In this study, we develop an efficient implementation of fill-reducing ordering for sparse direct linear solvers. Fill-reducing ordering is used to reduce the memory requirement and computational complexity in sparse direct solvers. Libraries such as METIS and ParMETIS that perform the nested dissection ordering are widely used. However, current available libraries are mainly optimized for CPU and are not necessarily optimized for GPU and Intel Xeon Phi. Another approach is the spectral nested dissection that recursively bipartitions the graph using the Fiedler vector (the eigenvector corresponding to the smallest non-zero eigenvalue) of the graph Laplacian. The Fiedler vector is usually computed by a sparse eigenvalue solver such as LOBPCG whose computational cost is dominated by the sparse matrix-vector multiplication (SpMV). Since many efforts for improving the performance of SpMV on GPU and Intel Xeon Phi have been made, the spectral nested dissection possibly outperforms existing libraries if we can adopt such efforts. In our study, we have implemented the spectral nested dissection for Intel Xeon Phi and have developed an efficient sparse matrix-vector multiplication by focusing on properties of the graph Laplacian. In this presentation, we show a performance comparison of our implementation of the spectral nested dissection and ParMETIS.

#74: **Supernodes ordering to enhance Block Low-Rank compression in sparse direct solvers**

Presenter: Mathieu Faverge
Co-authors:

In this talk, we present new ordering heuristics to perform block low-rank clustering in supernodes issued from the nested dissection. As kway partitioning within supernodes does not take into account interactions between supernodes, there is room to improve compression rates. We combine kway partitioning with a reordering strategy that aims at minimizing the number of off-diagonal blocks in the symbolic structure and show that both methods are limited. In addition, we propose a selection of some non-compressible vertices to handle the corresponding blocks in full-rank and reduce the burden on managing low-rank blocks with high ranks.

#75: **Complexity and parallelism of the solution phase in sparse direct solvers**

Presenter: Gilles Moreau
Co-authors: Alfredo Buttari, Jean-Yves L'Excellent, Théo Mary

The cost of the solution phase of sparse direct solvers, traditionally considered to be small with respect to that of the factorization phase, has become more important with the emergence of fast direct solvers based on low-rank approximations, and is even critical in presence of many right-hand sides. It therefore has become necessary to leverage the inherent differences of the solution phase compared to the factorization phase.
In this talk, we demonstrate that, because of its lower asymptotic complexity with respect to the factorization, the solve phase exhibits some interesting complexity and performance properties. We identify two algorithmic ingredients that bring a factor of acceleration that increases with the problem size and are thus critical to tackle large scale problems: namely, tree parallelism, and the possible sparsity of the right-hand sides. We also explain why these two ingredients are even more critical when used in conjunction with low-rank approximations such as the BLR or $\mathcal{H}$ formats. We illustrate these theoretical properties with some numerical experiments using the MUMPS solver on a set of large problems coming from a variety of real-life applications.

MS 3.2.B CAB G 61 PARALLEL-IN-TIME METHODS FOR HPC Chair: R. Krause

#76: **Challenges in solving turbulent flows with a purely time-periodic solver**

Presenter: Daniel Hupp
Co-authors: Dominik Obrist, Peter Arbenz

The continued increase in computational power is due to the availability of more processing units and not due to faster processing units. This lead to an increasing interest in methods that not only prallelize the space dimension

but also the time dimension. We have developed a solver for time-periodic Navier–Stokes problems. This is done by assuming a time-periodic steady-state solution and applying periodic boundary conditions in time. The resulting space-time problem is parallelized in space and time. The efficiency and good parallel scaling has been demonstrated in previous work.
In this talk, we show the performance of the solver applied to a turbulent flow. The considered flow is a periodically disturbed swept Hiemenz flow. This flow has been studied before, so we can localize the break down to turbulence precisely. We will show different challenges that have to be overcome to make the time-periodic solver work efficiently for turbulent flows.

#77: **Multigrid Reduction in Time (MGRIT) for Eddy Current Problems**

Presenter: Stephanie Friedhoff
Co-authors: Sebastian Schöps

Maxwell's equations are an essential tool in the numerical simulation of problems in electrical engineering. A standard approach for the simulation of electrical machines is to neglect the displacement current in Maxwell's equations, yielding the so-called magnetoquasistatic approximation or, synonymously, the eddy current problem. The computational complexity of classical solution algorithms based on a time-marching approach is high, particularly if long time periods have to be considered as, for example, in the case of simulating the start-up of an electrical machine. One approach for reducing the simulation time is with parallel-in-time integration techniques. In this talk, we consider Multigrid Reduction in Time (MGRIT) for the time-parallel solution of the eddy current problem. In particular, we present numerical results for a 2D model problem of a conducting wire surrounded by a pipe.

#78: **Analysis of Overlap in Optimized Waveform Relaxation Methods for RLCG Transmission Line Type Circuits**

Presenter: Pratik M. Kumbhar
Co-authors: Martin J. Gander, Albert E. Ruehli

Among many applications of parallel computing, solving large systems of ordinary differential equations (ODEs) which arise from large scale electronic circuits, or discretizations of partial differential equations (PDEs), form an important part. A systematic approach for their parallel solution are Waveform Relaxtion (WR) techniques, which were introduced in 1982 for circuit solver applications. These techniques are based on partitioning large circuits into smaller sub-circuits, which are then solved separately over multiple time steps, and the overall solution is obtained by an iteration between the sub-circuits. However, this technique can lead to non-uniform and potentially slow convergence over large time windows. To overcome this issue, optimized waveform relaxation techniques were introduced, which are based on optimizing a parameter. We show how this method improves the convergence for RLCG transmission line type circuits. We introduce overlap between sub circuits and analyze its effect on the convergence factor. For R=0, we find that these RLCG circuit equations represent discretizations of the well known Maxwell equations. We relate these two models and give some asymptotic results.

#79: **Parallel Solution of Time Dependent Problems using Non-Linear Multigrid Methods**

Presenter: Rolf Krause
Co-authors: Pietro Benedusi, Patrick Zulian, Carlo Garoni, Stefano Sera

We present a parallel and efficient multilevel solution strategy for solving non-linear time-dependent problems. We consider in particular the mono-domain model, a non-linear reaction-diffusion equation arising from a problem in electrophysiology: the electrical activation in the human heart. Different strategies for the space-time discretization and solution of the mono-domain equation are discussed, which are based on domain decomposition and multi-level methods. For the latter, we propose a semi-geo-metric multigrid method, for which the coarse level approximation spaces are created using arbitrary hierarchies of non-nested meshes. Interpolation and restriction in the multilevel context is then realized by means of a discrete L^2-projection between the non-matching meshes. This approach allows for creating the coarser levels of a multigrid hierarchy, even if only a single "fine" mesh is available. Hence, multigrid hierarchies can be created for arbitrary geometries in any dimension. We discuss how this approach can be applied to the monodomain equation discretised with space-time finite elements.
While we use continuous finite elements in space, for stability reasons we adopt discontinuous elements in time. We discuss shortly the properties of this time discretization scheme.
We investigate how different block smoothers, coarsening strategies and ordering of the space-time variables effect the overall convergence and robustness of the solver.
Furthermore, we comment on local time-stepping for space-time discretizations.
Finally, we investigate numerically the scalability and the convergence of our multilevel and domain decomposition solution strategies.

MS 3.2.C CAB G 51 RESILIENCE IN SCIENTIFIC COMPUTING Chair: L. Giraud

#80: **Soft error sensitivity of large scale CFD applications**

Presenter: E. Fatih Yetkin
Co-authors: Şenol Pişkin

Compute capabilities of largest High-performance computing (HPC) systems have increased by at least 100 times in the last 10 years and keep increasing substantially every year. This increase is made possible mostly by multi-core technology besides the increase in clock speed of CPUs. According to the literature, both are the main reasons for the increase of the possibility of bit-flip(s). Bit-flip(s) are defined as unexpected changes due to environmental situations on data during the calculation. Since there are systems with more than 100 thousand cores installed and available for processing simultaneously on these days, one can claim that the resiliency against bit-flip(s) is one of the key issues in computational science. On the other hand, computational simulation tools are always in need of more than available computational sources. This is the case for especially complex flow problems. A wide variety of natural virtue are classified as flow originated forces and modeled for predictions in many field: weather forecasting, aerodynamics, pharmaceutical design and biomedical engineering which spans from diagnosis, prognosis and pre-surgical planning to patient specific cardiovascular circulatory system and tumor treatment. There are many other diverse application areas, that can be described as flow problem. Moreover, the topics are not limited by the current problem definition since they are further extended with novel problem definitions. Computational fluid dynamics (CFD) is one of the most commonly used tool for solving these types problems. In this study, we are analyzing the reaction of a CFD simulator against to artificially generated transient soft errors at several phases of computation which are not impossible especially at peta/exa scale computing systems. First, the soft errors are induced into the system after the assembling the final global matrix of the simulator by manipulating predetermined bit flip operations. While the most time consuming part of the overall simulation is the linear solver it is the matrix-vector product for iterative matrix solvers like PCG or BiCG. Therefore, a random but non-zero element from the input vector of a randomly selected (in terms of iteration) matrix-vector product is manipulated by a bit-flip operation during the iterative matrix solver algorithm. Although the bit-flip manipulation is performed at randomly selected elements of the input vector of matrix-vector operations occurred during the calculation, it is performed in a systematic order such that the all of the sign, mantissa and exponential bits have been changed in order to test the sensitivity of the computations in bitwise level. Behavior of the CFD simulator is observed after iterative matrix solver and physical flow solution iterations. Soft error injection operation is repeated during the matrix vector multiplication operations on boundary conditions of the physical flow problem which corresponds to the right side of the (b vector) global matrix assembly equation. Results show that the iterative solvers of CFD matrices are highly sensitive to customized soft errors. Because of a soft error, even the physical system is stable and reliable, CFD solver may assess that the system has a non-physical solution. Hence, this study discussed an experimental framework to understand underlying conditions of the soft error(s) leading to unrealistic solutions while acceptable solutions should have been computed. As we present the very first preliminary results, further experiments and theoretical investigations are required for the real behavior of the flow simulator.

#81: **Some progresses in ULFM**

Presenter: George Bosilca
Co-authors:

#82: **Dynamic analysis of memory vulnerability**

Presenter: Sicong Zhuang
Co-authors: Marc Casas, Luc Jaulmes

Memory reliability is measured as a fault rate, i.e. a probability over a given amount of time. The missing link to know the fault probability of any data stored in memory is its storage duration. By analyzing memory access patterns of an application, we can thus determine the vulnerability of data stored in memory, and thus the optimal amount of redundancy to keep fault probabilities below an acceptable threshold at all times. This allows to dynamically get fault probabilities for memory storage, and opens the door to runtime optimizations. The open problem remains the right set of actuators to use for a runtime system, in order to adapt the strength of memory protection. Some leads are to have different ECC strengths, either through an adaptable ECC scheme whose amount of redundancy can be adjusted or through different chips with the option of migrating data under different protection requirements.

#83: **Using CRAFT library to introduce fault tolerance in PHIST iterative algorithms**

Presenter: Faisal Shahzad
Co-authors: Jonas Thies, Moritz Kreutzer, Thomas Zeiser, Georg Hager, Gerhard Wellein

In this talk, we will present our fault tolerance approach for sparse linear algebra solvers. The talk consists of two components. In the first part, we will present our fault tolerance solution in the form of CRAFT (Checkpoint/Restart and Automatic Fault Tolerance) library, whereas its usage and benchmarks will be discussed in the second part. The CRAFT library serves two fault tolerance functions: 1) Checkpoint/Restart (CR): This com-

ponent is intended to reduce the programmer's effort for incorporating the application-level CR functionality in libraries and user-applications. It supports a range of default data-types (e.g., plain-old-data types (POD), 1D- and 2D-POD arrays, MPI derived data-types, etc.), which can be used directly out-of-the-box with minimal code-changes. For each of these POD data elements, the user can opt between various IO formats, e.g., MPI-IO, Binary/ASCII-format. In addition to these default data-types, the user can extend the CRAFT library to checkpoint any arbitrary data-types. As overhead reduction strategies, the CR-component has a built-in asynchronous checkpointing mechanism for PFS-level checkpoints and also supports the usage of the SCR (Scalable Checkpoint/Restart) library for node-level checkpoints. Its signal-based checkpointing feature can be used to initiate checkpoints at any desired time from the terminal instead of setting a fixed frequency in the application. In addition, the CR-component supports multiple- and nested-checkpoints. Hereby, nested checkpoints (at different nested loops/stages of an application) require careful attention as they can result in data inconsistency. For this purpose, the concept of child-parent checkpoints is introduced. 2) Automatic Fault Tolerance (AFT): This component of CRAFT makes use of ULFM-MPI to provide an easier interface for a dynamic recovery in case of process failures. The ULFM-MPI is a prototype fault tolerance MPI implementation that provides necessary routines to detect, acknowledge, and recover from process failures. However, the actual communication and application recovery strategy is not a generic task and therefore is left up to the user to design and implement in the application. CRAFT's AFT component hides the details of these ULFM-MPI communication recovery functionalities by providing a simpler interface which could be applied to a large variety of applications. It supports shrinking and non-shrinking communication recoveries. In case of a shrinking recovery, the AFT functions offer all necessary information about the failure to devise a recovery operation for failed processes. AFT also manages the resource information about the application and job. Thereby the user can allocate reserve nodes which could be used for recovery processes in case of a non-shrinking recovery. Although both parts of CRAFT complement each other, they can still be used independently. In the second part of the talk, we will discuss the usage of CRAFT to introduce fault tolerance in the PHIST (Pipelined Hybrid Parallel Iterative Solver Toolkit) library algorithms. The PHIST library provides implementation of and interfaces to block iterative solvers for sparse linear and eigenvalue problems and supports multiple backends, e.g., Trilinos, PETSc, GHOST, etc. We will present how we used and extended the CRAFT library to support PHIST data-types that are necessary for checkpointing. We will then show the usage of CRAFT to introduce fault tolerance in PHIST-based Lanczos and Jacobi-Davidson algorithms. Using these implementations we first analyze the overheads involved by CRAFT's checkpoint/restart mechanism and its optimizations. Secondly we also investigate different overhead components for dynamic process recovery mechanism in the application.

Giuseppe **Accaputo**	ETH Zürich, Switzerland
Emmanuel **Agullo**	INRIA Bordeaux, France
Hussam **Al Daas**	INRIA, France
Peter **Arbenz**	ETH Zurich, Switzerland
Hilal **Arslan**	Roketsan Missiles Inc., Turkey
Zhong-Zhi **Bai**	Chinese Academy of Sciences, China
Achim **Basermann**	German Aerospace Center (DLR), Germany
Martin **Bečka**	SAS Bratislava, Slovakia
Costas **Bekas**	IBM Research - Zurich, Switzerland
Matthias **Bollhöfer**	TU Braunschweig, Germany
George **Bosilca**	University of Tennessee, Knoxville, United States
Nela **Bosner**	University of Zagreb, Croatia
Bérenger **Bramas**	MPCDF, Germany
Alfredo **Buttari**	CNRS-IRIT-INPT, France
Leopold **Cambier**	Stanford University, United States
Carmen **Campos**	U. Politecnica Valencia, Spain
Christian **Carbogno**	Fritz-Haber-Institut der Max-Planck-Gesellschaft, Germany
Rocío **Carratalá-Sáez**	Universitat Jaume I, Spain
Cedric **Chevalier**	CEA, France
Siegfried **Cools**	University of Antwerp, Belgium
Jeffrey **Cornelis**	University of Antwerp, Belgium
Edoardo Angelo **Di Napoli**	Forschungszentrum Juelich, Germany
Jingqiu **Ding**	ETH Zurich, Switzerland
Mathieu **Faverge**	Bordeaux INP - Inria, France
Thomas **Forell**	CST GmbH, Germany
Simon **Frasch**	ETH Zurich, Switzerland
Matthias **Frey**	PSI, Switzerland
Stephanie **Friedhoff**	University of Wuppertal, Germany
Yasunori **Futamura**	University of Tsukuba, Japan
Efstratios **Gallopoulos**	University of Patras, Greece
Wilfried **Gansterer**	University of Vienna, Austria
Dan **Gordon**	University of Haifa, Israel
Stef **Graillat**	Sorbonne University, France
Inge **Gutheil**	Forschungszentrum Juelich, Germany
Vaclav **Hapla**	ETH Zurich, Switzerland
Hidehiko **Hasegawa**	University of Tsukuba, Japan
David **Horak**	VSB-Technical University of Ostrava, Czech Republic
Thomas **Huckle**	Technical University of Munich, Germany
Daniel **Hupp**	ETH Zurich, Switzerland
Roman **Iakymchuk**	KTH Royal Institute of Technology, Sweden
Toshiyuki **Imamura**	RIKEN Center for Computational Science, Japan
Yuta **Inagawa**	University of Tsukuba, Japan
Jun-Ichi **Iwata**	Advance Soft Corporation, Japan
Filip **Janicki**	ETH Zurich, Switzerland
Zakariae **Jorti**	IFPEN, France
Wilfried **Kirschenmann**	ANEO, France
Timon **Knigge**	Utrecht University, Switzerland
Marija **Kranjcevic**	ETH Zurich, Switzerland
Rolf **Krause**	USI - Università della Svizzera italiana, Switzerland
Daniel **Kressner**	EPFL, Switzerland
Jakub **Kruzik**	Institute of Geonics of the CAS, Czech Republic
Pratik Mahadeo **Kumbhar**	University of Geneva, Switzerland
Lahcen **Laayouni**	Al Akhawayn University, Morocco
Bruno **Lang**	University of Wuppertal, Germany
Alfio **Lazzaro**	University of Zurich, Switzerland
Florent **Lopez**	Rutherford Appleton Laboratory, United Kingdom
Hatem **Ltaief**	KAUST ECRC, Saudi Arabia
Andreas **Mang**	University of Houston, United States
Murat **Manguoglu**	TU Berlin and METU (TR) , Germany
Valeriy **Manin**	Bergische Universität Wuppertal, Germany
Svetozar **Margenov**	IICT - BAS, Bulgaria
Guoyan **Meng**	Xinzhou Teachers University, China
Salli **Moustafa**	ANEO, France

Frederic **Nataf**	INRIA, France
Esmond **Ng**	Berkeley Lab, United States
Vedran **Novaković**	Universitat Jaume I, Spain
Takeshi **Ogita**	Tokyo Woman's Christian University, Japan
Gabriel **Oksa**	SAS Bratislava, Slovakia
Katsuhisa **Ozaki**	Shibaura Institute of Technology, Japan
Dario **Pasquini**	La Sapienza, IAC-CNR Rome, Italy
Marek **Pecha**	VSB-TUO, Czech Republic
Eric **Polizzi**	UMass Amherst, United States
Michael **Rippl**	TU München, Germany
Jose E **Roman**	U. Politecnica Valencia, Spain
Eloy **Romero Alcalde**	College of William and Mary, United States
Yousef **Saad**	University of Minnesota, United States
Olaf **Schenk**	USI - Università della Svizzera italiana, Switzerland
Nick **Schenkels**	University of Antwerp, Belgium
Faisal **Shahzad**	Universität Erlangen-Nürnberg, Germany
Sanja **Singer**	University of Zagreb, Croatia
Saša **Singer**	University of Zagreb, Croatia
Jacob **Snoeijer**	University of Antwerp, Belgium
Aleksandros **Sobczyk**	IBM Research – Zurich, Switzerland
Edgar **Solomonik**	University of Illinois at Urbana-Champaign, United States
Andreas **Stathopoulos**	College of William and Mary, United States
Masaki **Suwa**	Kogakuin University, Japan
Guillaume **SYLVAND**	Airbus CRT / Inria, France
Taekshi **Terao**	SIT, Japan
Jonas **Thies**	German Aerospace Center (DLR), Germany
Marian **Vajtersic**	University of Salzburg, Austria
Rui-Ping **Wen**	Taiyuan Normal University, China
Michiel **Wouters**	University of Antwerp, Belgium
Xinzhe **Wu**	CNRS/MDLS and U. Lille, France
Chao **Yang**	Lawrence Berkeley National Laboratory, United States
Emrullah Fatih **Yetkin**	Kadir Has University, Turkey
Wenzhe **Yu**	Duke University, United States
Guo-Feng **Zhang**	Lanzhou University, China
Sicong **Zhuang**	BSC, Spain
Mawussi **Zounon**	University of Manchester, United Kingdom
Ian **Zwaan**	Universität Wuppertal, Germany