Sparse Days

June 30th-July 1st, 2016

CERFACS, Toulouse, France

Sparse Days - June 30th-July 1st, 2016 - CERFACS

Thursday, June 30th 2016

10.30 - 11.00 Registration and welcome coffee

Session I

11.00 - 11.30 *Sparse Linear Algebra Support in Intel Math Kernel Library*
A. Kalinkin (Intel, Novosibirsk, Russia)

11.30 - 12.00 *Overview of Intel Math Kernel Library (Intel MKL) Solutions for Eigenvalue Problems*
I. Sokolova (Intel, Novosibirsk, Russia)

Lunch

Session II

14.00 - 14.30 *Block Iterative Methods and Recycling for Improved Scalability of Linear Solvers*
P. Jolivet (IRIT, Toulouse, France)

14.30 - 15.00 *Coarse Grid Correction for the MaPHyS Algebraic Domain Decomposition Solver*
L. Poirel (INRIA Bordeaux Sud-Ouest, Bordeaux, France)

15.00 - 15.30 *Geometric, Algebraic, and Graph Issues of DDM*
V. Il'in (Novosibirsk State University, Novosibirsk, Russia)

15.30 - 16.00 Coffee break

Session III

16.00 - 16.30 *Overview of Task-based Sparse and Data-sparse Solvers on Top of Runtime Systems*
E. Agullo (INRIA Bordeaux Sud-Ouest, Bordeaux, France)

16.30 - 17.00 *Task-based Sparse Cholesky Solver on Top of Runtime System*
F. Lopez (Rutherford Appleton Laboratory, UK)

19.30 Banquet at the restaurant "Côté Garonne" in Toulouse.

Sparse Days - June 30th-July 1st, 2016 - CERFACS

Friday, July 1st 2016

Session IV

9.00 - 9.30 *The Remarkable Accuracy of the Lanczos Process*
C. Paige (McGill University, Montreal, Canada)

9.30 - 10.00 *PFEAST: A High Performance Eigenvalue Solver using Three Full Levels of MPI Parallelism*
E. Polizzi (University of Massachusetts, Amherst, USA)

10.00 - 10.30 *The Challenge of Large Sparse Rank Deficient Least Squares Problems*
J. Scott (Rutherford Appleton Laboratory, UK)

10.30 - 11.00 Coffee break

Session V

11.00 - 11.30 *One-pass Substitution in the Left-looking LU Factorization*
R. Luce (EPFL, Lausanne, Switzerland)

11.30 - 12.00 *On the Complexity of the Block Low-Rank Multifrontal Factorization*
T. Mary (IRIT, Toulouse, France)

12.00 - 12.30 *Scheduling Sparse Symmetric Fan-Both Cholesky Factorization*
M. Jacquelin (Lawrence Berkeley National Laboratory, USA)

Lunch

Session VI

14.00 - 14.30 *Numerically-aware Nested Dissection Ordering*
J. Hogg (Rutherford Appleton Laboratory, UK)

14.30 - 15.00 *Hierarchical Probing for General Sparse Matrices, a Method for Computing diag($f(A)$)*
A. Stathopoulos (College of William and Mary, Williamsburg, USA)

15.00 - 15.30 *High Performance Matrix-matrix Multiplication of Very Small Matrices*
I. Masliah (LRI, Orsay, France)

Closure

June 30th - Session I

Sparse Linear Algebra Support in Intel Math Kernel Library

A. Kalinkin (Intel, Novosibirsk, Russia).

This talk is devoted to the functionality provided by Intel Math Kernel Library (Intel MKL) to support Linear Algebra computations with sparse data. Modern multi-/many-core architectures have certain limitations and benefits for computations with sparse data. Depending on the type of the problem to solve, it is still possible to find a smart way to minimize the impact of limitations and explore benefits to their full potential so that underlying computer architecture capable to solve huge Linear Algebra problems. The talk covers the tools for basic manipulations with matrices and vectors needed for more complex computations like those conducted in iterative solvers, advanced approaches needed for direct solvers to work in a fast and memory efficient way, and eigenvalue & eigenvector problem solvers demonstrating robust numerical behavior.

Overview of Intel Math Kernel Library (Intel MKL) Solutions for Eigenvalue Problems

I. Sokolova (Intel, Novosibirsk, Russia).

We outline both existing and prospective Intel Math Kernel Library (Intel MKL) solutions for standard and generalized Hermitian eigenvalue problems. Extended Eigensolver functionality, based on the accelerated subspace iteration FEAST algorithm, is a high-performance solution for obtaining all the eigenvalues and optionally all the associated eigenvectors, within a user-specified interval. To extend the available functionality, a new approach for finding the k largest or smallest eigenvalues is proposed. We envision automatically producing a search interval by a preprocessing step based on classical and recent methods that estimate eigenvalue counts. Computational results are presented.

June 30th - Session II

Block Iterative Methods and Recycling for Improved Scalability of Linear Solvers

P. Jolivet (IRIT, Toulouse, France).

On the one hand, block iterative methods may be useful when solving systems with multiple right-hand sides, for example when dealing with time-harmonic Maxwells equations. They indeed offer higher arithmetic intensity, and typically decrease the number of iterations of Krylov solvers. On the other hand, recycling also provides a way to decrease the time to solution of successive linear solves, when all right-hand sides are not available at the same time. I will present some results using both approaches, as well as their implementation inside the open-source framework HPDDM (https://github.com/hpddm/hpddm)

Coarse Grid Correction for the MaPHyS Algebraic Domain Decomposition Solver

L. Poirel (INRIA Bordeaux Sud-Ouest, Bordeaux, France).

Over the last few decades, there have been innumerable science, engineering and societal breakthroughs enabled by the development of High Performance Computing (HPC) applications, algorithms and architectures.

In the context of this talk, our focus is on numerical linear algebra algorithms that appear in many large scale simulations and are often the most time consuming numerical kernel; more precisely we consider numerical schemes for the solution of large sparse systems of linear equations.

This presentation will focus on hybrid methods that hierarchically combine direct and iterative methods for the solution of large sparse systems of linear equations. These techniques

inherit the advantages of each approach, namely the limited amount of memory and natural parallelization for the iterative component and the numerical robustness of the direct part.

In order to perform extreme scale simulations on large distributed platforms, the number of iterations is often the main limitation of these methods since the convergence may deteriorate with the number of computational nodes. Taking into account a low-rank approximation of our problem, we are able to alleviate this penalizing numerical effect.

Focusing on the distributed sparse hybrid solver MaPHyS originated at CERFACS and further expanded and developed in the HiePACS Inria team, we will show how the adaptation of a semi-algebraic coarse space originally proposed in the context of domain decomposition methods ensures the scalability of a distributed sparse hybrid solver.

Geometric, Algebraic, and Graph Issues of DDM

V.Il'in (Institute of Computational Mathematics and Mathematical Geophysics, SB RAS and Novosibirsk State University, Novosibirsk, Russia).

Various approaches of the parallel domain decomposition methods to solve ill-conditioned sparse large SLAEs which arise in the FEM or FVM approximations of the multi-dimensional boundary value problems on non-structured meshes are considered. A scalable parallelism is provided via hybrid programming by means of MPI-processes and multi-thread computing on heterogeneous multi-core clusters with distributed and hierarchical shared memory. Geometric DDMs are based on the balancing decomposition of the grid computational domain into its subdomains with or without a parameterized overlapping. Different interface conditions on the internal boundaries of adjacent subdomains, as well as some special grid skeleton structures are used for an optimization of the additive Schwarz iterative procedure with synchronized solving of auxiliary subdomain SLAEs by a direct or an iterative solvers.

The distributed multi-preconditioned semi-conjugate residual method in the block Krylov subspaces is employed at the upper level of two-level iterative process, which is accelerated by a coarse grid correction with low rank approximation of the original matrix using the different order basis functions in the Galerkin concept. The algebraic decomposition is implemented by some graph partition techniques. A minimization of communicational costs is provided by construction of the exchange buffers and by a special schedule of the data exchange between subdomains.

The algorithms proposed are implemented in KRYLOV library without any limitations on the degree of freedom and on the number of the computational nodes. Results of some preliminary numerical experiments for the model grid boundary value problems are presented.

Joint work with Y. L. Gurieva (Institute of Computational Mathematics and Mathematical Geophysics, SB RAS, Russia).

June 30th - Session III

Overview of Task-based Sparse and Data-sparse Solvers on Top of Runtime Systems

E. Agullo (INRIA Bordeaux Sud-Ouest, Bordeaux, France).

The complexity of the hardware architectures of modern supercomputers led the community of developers of scientific libraries to adopt new parallel programming paradigms. Among them, task-based programming has certainly become one of the most popular as it allows for high productivity while ensuring high performance and portability by delegating tasks management to a runtime system. In this talk, we will present an overview of sparse solvers that have been designed in the context of the Matrices Over Runtime Systems @ Exascale (MORSE) and Solvers for Heterogeneous Architectures (SOLHAR) projects. We will present the design of new direct solvers implementing supernodal (PaStiX) and multifrontal (qr_mumps) methods, new Krylov solvers ensuring pipelining both at a numerical and software level, new sparse hybrid methods

(MaPHyS) as well as data sparse libraries implementing fast multipole methods (ScalFMM) and hierarchical matrices (hmat, in collaboration with Airbus Group Innovations). For all these methods, we will highlight the challenges we have faced in terms of expressivity, granularity, scheduling and scalability and illustrate their performance on large academic and industrial test problems.

Task-based Sparse Cholesky Solver on Top of Runtime System

F. Lopez (Rutherford Appleton Laboratory, UK).

In this talk we present the implementation of a task-based sparse Cholesky solver on top of runtime system. To achieve this, we use two different programming models: a Sequential Task Flow (STF) model and Parametrized Task Graph (PTG) model. We first present an STF-based implementation of our code using both the StarPU runtime system and the OpenMP 4.0 standard and then, we move to a PTG model using the PaRSEC runtime system. We compare these implementations against the state-of-the-art MA87 solver from the HSL library to assess our approach in terms of performance and scalability on shared-memory multicore architectures.

July 1st - Session IV

The Remarkable Accuracy of the Lanczos Process

C. Paige (McGill University, School of Computer Science, Montreal, Canada).

Cornelius Lanczos's 1952 process for tridiagonalizing a symmetric matrix A is the basis for several very useful large sparse matrix algorithms. Even Golub and Kahan's 1965 bidiagonalization of general possibly non-square A can be formulated as a Lanczos process. We know that the finite precision process can lose orthogonality immediately the first eigenvector of A has converged to machine precision, yet we still obtain accurate results. Here we will prove this, and show that its behaviour is really quite beautiful, especially for an algorithm which until about 1971, many believed did not work.

After k steps, the process on a machine with floating point precision ϵ can be modelled as

$$AV_k = V_k T_k + v_{k+1}\beta_{k+1}e_k^T + E_k, \qquad V_k = [v_1, \ldots, v_k], \qquad \|E_k\|_2 \le O(\epsilon)\|A\|_2,$$

where $T_k = T_k^T$ is the computed tridiagonal, and the v_j are the computed vectors normalized (in theory) to have $\|v_j\|_2 = 1$. This leads to the crucial matrix S_k, where S_k and U_k are strictly upper triangular:

$$V_k^T V_k = U_k^T + I_k + U_k, \qquad S_k = (I + U_k)^{-1}U_k, \qquad (I + U_k)(I - S_k) = I_k.$$

No singular value of S_k is greater than one, and each unit singular value corresponds to a loss of rank in V_k. All the eigenvalues of S_k are zero. Both the singular value decomposition (SVD) and Jordan canonical form (Jcf) of S_k are used in revealing the beautiful behaviour of the Lanczos process. This knowledge should lead to greater confidence in the use of the many large sparse matrix methods based on the Lanczos process or the Golub-Kahan bidiagonalization (GKB), and improved algorithms.

PFEAST: A High Performance Eigenvalue Solver using Three Full Levels of MPI Parallelism

E. Polizzi (University of Massachusetts, Amherst, USA).

The FEAST algorithm and eigensolver for interior eigenvalue problems naturally possesses three distinct levels of parallelism. The solver is then suited to exploit modern computer architectures containing many interconnected processors. This presentation highlights a recent development within the software package that allows the dominant computational task, solving a set of complex linear systems, to be performed with a distributed-memory solver. The software, written with a reverse-communication-interface, can now be interfaced with any generic MPI linear-system solver using a customized data distribution for the eigenvector solutions. This

work utilizes two common black-box distributed-memory linear-systems solvers (Cluster- MKL-Pardiso and MUMPS), as well as our own application-specific domain-decomposition MPI solver, for a collection of 3-dimensional finite-element systems in electronic structure calculations. We discuss and analyze how parallel resources can be placed at all three levels simultaneously in order to achieve good scalability and optimal use of the computing platform.

The Challenge of Large Sparse Rank Deficient Least Squares Problems

J. Scott (Rutherford Appleton Laboratory, UK).

We are interested in solving the least squares problem

$$min\|Ax - b\|_2$$

where the $m \times n, (m \geq n)$ matrix A is large, sparse and rank deficient. Such problems arise in a number of practical applications and are tough to solve reliably and efficiently. In this talk, we focus on methods based on Cholesky-based factorizations. By examining the performance of modern parallel sparse direct solvers and exploiting our knowledge of the algorithms behind them, we perform numerical experiments to study how they can be used to efficiently solve these problems. We consider both the regularized normal equations and the regularized augmented system. We employ the computed factors of the regularized systems as preconditioners with an iterative solver to obtain the solution of the original (unregularized) problem. Furthermore, we look at using limited-memory incomplete Cholesky-based factorizations and how these can offer the potential to solve very large problems.

July 1st - Session V

One-pass Substitution in the Left-looking LU Factorization

R. Luce (EPFL, Lausanne, Switzerland).

The Gilbert-Peierls variant of the sparse LU factorization computes the columns of L and U in a left looking manner. At each stage of the factorization two graph traversals of the intermediate L factor are performed: The first determines an ordering for a following forward substitution with L, while during a second pass through the graph of L the actual floating point operations are carried out. By maintaining an auxiliary data structure that captures the dependency structure of the intermediate factors, we show that the two steps can actually be combined in one single graph traversal.

On the Complexity of the Block Low-Rank Multifrontal Factorization

T. Mary (IRIT, Toulouse, France).

Low-rank compression relying on the Block Low-Rank format (BLR) has been shown to provide significant gains compared to full-rank on practical applications requiring the solution of large sparse systems of linear equations. However, unlike hierarchical formats, such as H and HSS, its theoretical complexity was unknown. We extend the theoretical work done on hierarchical matrices in order to compute the theoretical complexity of the BLR multifrontal factorization and we present several variants of the BLR multifrontal factorization. We then show how these variants can further reduce the complexity of the factorization and we provide an experimental study with numerical results to support our complexity bounds.

Scheduling Sparse Symmetric Fan-Both Cholesky Factorization

M. Jacquelin (Lawrence Berkeley National Laboratory, USA).

In this work, we study various scheduling approaches for Sparse Cholesky factorization and introduce a new solver, symPACK. We review different algorithms based on a distributed memory implementation of the Fan-Both algorithm. We also study different communication mechanisms and dynamic scheduling techniques.

July 1st - Session VI

Numerically-aware Nested Dissection Ordering

J. Hogg (Rutherford Appleton Laboratory, UK).

For numerically difficult sparse symmetric systems such as those often arising in interior point methods, sometimes the application of a scaling alone in the preprocessing step is insufficient to make the problem numerically tractable for a direct solver to solve without significant performance-inhibiting pivoting.

In such cases the enforcement of specific 2×2 pivots through the use of a restricted ordering is often effective, such as the method based on MC64 suggested by Duff and Pralet. However such restrictions are normally crude and result in significantly more fill that an unrestricted ordering.

In this talk we will describe a modified nested dissection method that takes into account numerical values of the matrix and avoids separators likely to result in delayed pivots. We will present results demonstrating that it can keep the number of delayed pivots in a subsequent LDL^T factorization low, being almost as numerically effective as existing methods, but with significantly less additional fill.

Hierarchical Probing for General Sparse Matrices, a Method for Computing $diag(f(A))$

A. Stathopoulos (College of William and Mary, Williamsburg, USA).

A problem that occurs frequently in numerical linear algebra is the computation of the diagonal of a function of a matrix A. In particular, many applications such as Lattice Quantum Chromodynamics (LCQD), data mining, statistics, and uncertainty quantification require the computation of the trace of the inverse of A. For very large, sparse matrices, methods based on factorization are not practical and stochastic (Monte Carlo) approaches have become the norm. Still, deterministic sparse matrix methods can help reduce the variance of the Monte Carlo.

Probing is such a method. For many matrices, the magnitude of the elements of $f(A)_{i,j}$ decrease inversely with the graph theoretical distance between nodes i, j of A. Therefore, if we compute a k-distance coloring of A (or equivalently the coloring of A^k), then the values of the nodes that share the same color can be recovered by creating a probing vector consisting of all ones for nodes sharing the same color, and zeros everywhere else. Thus, $f(A)$ has to be applied only to n vectors, where n is the number of colors used to color A^k. There are two problems with this approach: (1) the memory and computational time required to compute and store A^k increases intractably with k, (2) if a certain k gives the trace with low accuracy, repeating with a higher k means that all previous solves have to be discarded since the intersection between sets of probing vectors for different k is likely to be empty.

In our research we address these problems by first providing approximate ways to perform the k-distance coloring of A and second by enforcing a hierarchy of colorings so that the work done at some k will be a subset of the work needed for larger k. For the regular lattices of LQCD we developed algorithms to perform both of these tasks inexpensively and elegantly. For general matrices, we address the problem in a multi-level way.

Our approach borrows from the graph coarsening ideas in METIS and of Walshaw's multilevel algorithm, but for large distance coloring. Our goal is to identify groups of nodes that are as weakly connected as possible. At any level i, we first distance-1 color the sparse graph

$A(i)$. Then, for each node of $A(i)$ we merge it with another node that is distance-2 away. The neighborhood of the new node in the coarse graph at level $i+1$ will be the union of the neighborhoods of the two level i nodes. We stop when the coarsest $A(i)$ graph is of trivially small size. Using the resulting hierarchy of colors as a mixed radix basis, we produce a hierarchical sequence of colors and thus of probing vectors, with the property that at level i all distance-2^i error is annihilated.

We will discuss the recursive algorithm and the various choices of colorings (both discrete and spectral) that can be used at each level. As our experiments show, the probing vectors have the same effect as classical probing (as expected) but with tractable execution time and storage.

High Performance Matrix-matrix Multiplication of Very Small Matrices

I. Masliah (LRI, Orsay, France).

The use of the general dense matrix-matrix multiplication (GEMM) is fundamental for obtaining high performance in many scientific computing applications. However, GEMMs for small matrices (of sizes less than 32) can be further optimized in existing libraries. We consider the case of many small GEMMs on either CPU or GPU architectures. This case often occurs in applications such as high-order Finite Element Methods, Multifrontal QR factorization and others. The GEMMs are grouped together in a single batched routine. We present specific algorithms for these cases and optimization techniques allowing to obtain a performance that can be within 90% of the peak performance. We show that these results outperform available state-of-the-art implementations and vendor-tuned math libraries.

Last update: June 28, 2016