Scalable Direct Solvers for Electromagnetics and Multiphysics Simulations

Description:

This 4-hour short course provides an overview of state-of-the-art direct solvers and preconditioners for large-scale linear systems for integral equation, finite-difference and finite-element matrices arising in electromagnetic and multiphysics simulations. These techniques provide an appealing avenue for increasing capability and efficiency of existing simulators and the effectiveness of computational researchers for solving complex problems. The course also introduces a variety of open-source software packages developed at Lawrence Berkeley National Laboratory (LBNL). Audience may bring their laptops to participate the hands-on sessions.

Large-scale and full-wave modeling, analysis and synthesis of electromagnetic and multiphysics systems for traditional and emerging RF, microwave, terahertz applications rely on efficient numerical tools. Such physical systems, after discretization, lead to linear, linearized, and/or coupled systems of equations. These numerical systems can be solved by either iterative or direct solvers. Iterative solvers, despite their success in constructing well-conditioned formulations and fast multipole-type algorithms, remain inefficient for systems that are inherently ill-conditioned and/or require multiple right-hand sides. This is particularly true for design automation, inverse scattering, and ill-conditioned coupled systems of different physics where iterative solvers often require forbiddingly high computational time. Direct solvers, in stark contrast, can attain reliable solutions in a predictable time. Fast direct solvers rely on the fact that off-diagonal blocks of the well-ordered linear systems can be compressed by numerical linear algebra tools including low-rank and butterfly decompositions. When further embedded in hierarchal matrix frameworks, such as H-matrix, hierarchically off-diagonal low-rank (HODLR), block low-rank (BLR) and hierarchically semi-separable (HSS) formats, these solvers can achieve quasi-linear complexities for construction, factorization and solution of large-scale dense and sparse systems. In addition, these techniques also allow construction of very efficient algebraic preconditioners.

Despite numerous papers published in literature, the development of these advanced tools and their distributed-memory implementations require a considerable amount of time, oftentimes deter practitioners and computational researchers from developing highly efficient linear solvers. The lack of robust software prevents these advanced mathematical tools from being used in the application domains The Scalable Solvers Group at LBNL has been working to bridge the gap between the theory and practice. We have developed several open-source mathematical software packages, including SuperLU_DIST—a supernodal solver for large sparse systems, STRUMPACK—a multi-frontal solver enhanced with hierarchical matrix techniques for large sparse and dense systems, and ButterflyPACK—a low-rank and butterfly enhanced hierarchical matrix solver for large dense systems. The availabilities of these packages greatly reduce the development difficulties for advanced integral equation, finite element, finite difference, and coupled numerical simulators.

Course Object:

The audience will get familiar with various compressed representations of dense and sparse matrices for efficient storage and operations. The audience will learn how to use open-source distributed-memory software packages, namely, SuperLU_DIST, STRUMPACK and ButterflyPACK to solve large-scale sparse and dense linear systems.

Course Outline:
The proposed outline consists of two sessions with focuses on dense and sparse linear solvers, respectively.

- Session I: fast direct solvers for integral equations
  - overview of iterative, direct and preconditioned linear solvers
  - analysis of Green's function
  - numerical linear algebra tools: low-rank and butterfly decompositions
  - hierarchical matrix formats: H matrix, HODLR, BLR, and HSS, and their randomized construction, factorization and solution
  - hands-on: STRUMPACK and ButterflyPACK for integral equation systems
- Coffee break
- Session II: fast direct solvers for differential equations
  - introduction to sparse matrix formats
  - supernodal and multi-frontal sparse solvers
  - hierarchical matrix compressions for frontal matrices and their randomized construction, factorization and solution
  - hands-on: SuperLU_DIST and STRUMPACK for finite-element systems

Instructors:
Yang Liu is a postdoctoral fellow in the Scalable Solvers Group of the Computational Research Division at Lawrence Berkeley National Laboratory, in Berkeley, California. Dr. Liu received the Ph.D. degree in electrical engineering from the University of Michigan in 2015. From 2015 to 2017, he worked as a postdoctoral fellow at the Radiation Laboratory, University of Michigan. His main research interest is in computational electromagnetics (including fast time-domain integral equation solvers, fast direct integral and differential equation solvers, and multi-physics modeling), numerical linear and multi-linear algebras (including sparse solvers, randomized low-rank, butterfly and tensor algebras), and high-performance scientific computing. Dr. Liu authored and co-authored the Sergei A. Schelkunoff Transactions Prize Paper, APS 2018, second place student paper, ACES 2012, and the first place student paper, FEM 2014.

Pieter Ghysels is a research scientist in the Scalable Solvers Group of the Computational Research Division at Lawrence Berkeley National Laboratory, in Berkeley, California. His main interests are in High Performance Computing (HPC) and linear algebra. Pieter has expertise in both iterative methods and direct methods for the solution of systems of linear equations. He is the main developer of the STRUMPACK software library which offers a direct solver and preconditioners for large sparse linear systems as well as memory efficient representations of structured dense matrices. Pieter Ghysels received an engineering degree (in 2006) and completed a PhD in engineering Sciences, both at the (Flemmish) Catholic University in Leuven, Belgium. From 2010-2013, Pieter worked at the Universiteit Antwerpen (University of Antwerp, Belgium) and at the Intel Exascience Lab Flanders.