SLAC-PUB-12984
November 2007

Enabling technologies for petascale electromagnetic accelerator simulation*

Lie-Quan Lee, Volkan Akcelik, Sheng Chen, Lixin Ge, Ernesto Prudencio, Greg Schussman, Ravi Uplenchwar, Cho Ng, and Kwok Ko

SLAC, 2575 Sand Hill Road, M/S 27, Menlo Park, CA 94025

Xiaojun Luo and Mark Shephard

Rensselar Polytechnic Institute, 110 8th Street, Troy, NY 12180

SciDAC CET/Institute Collaborators: E. Ng, X. Li, C, Yang, P. Husbands, A. Pinar, D. Bailey, D. Gunter (LBNL); L. Diachin, D. Brown, D. Quinlan, R. Vuduc (LLNL); M. Shephard, X. Luo, A. Brewer (RPI); P. Knupp, K. Devine, E. Boman, J. Kraftcheck (SNL); O. Ghattas (UT Austin); D. Keyes (Columbia); K. Ma, Z. Bai, B. Liao (UC Davis); T. Tautges (ANL); R. Barrett (ORNL)

liequan@slac.stanford.edu

Abstract. The SciDAC2 accelerator project at SLAC aims to simulate an entire three-cryomodule radio frequency (RF) unit of the International Linear Collider (ILC) main Linac. Petascale computing resources supported by advances in Applied Mathematics (AM) and Computer Science (CS) and INCITE Program are essential to enable such very large-scale electromagnetic accelerator simulations required by the ILC Global Design Effort. This poster presents the recent advances and achievements in the areas of CS/AM through collaborations.

1. Introduction

The International Linear Collider (ILC) is the highest priority mid-term HEP facility listed in DOE report "Facilities for the Future of Science: A Twenty-Year Outlook". The COMPASS accelerator project is charged by the ILC Global Design Effort to simulate an entire three-cryomodule RF unit of the ILC main Linac with realistic dimensions and misalignment. Large-scale electromagnetic modeling at petascale supercomputers, together with advances through collaborations with SciDAC CET's/Institutes in the areas of CS/AM, is essential to achieving this goal. An integrated effort in developing parallel eigensolvers, linear solvers, optimization algorithms, adaptive refinements and visualization techniques has been established. In the following, we report several recent major advances.

2. Determining deformed shape of ILC cavities

Accelerating cavities for International Linear Collider (ILC) are manufactured with loose machining tolerance, so they need to be tuned to the accelerating frequency of 1.3GHz. This process leads to shape changes on the order of 100 microns from the designed shape. The deformed cavity can significantly change High-Order-Mode (HOM) damping and hence the beam emittance.

Contributed to 7th International Conference on Symmetry in Nonlinear Mathematical Physics, 06/24/2007--6/30/2007, Kyiv, Ukraine

Work supported in part by US Department of Energy contract DE-AC02-76SF00515

Collaborating with scientists from TOPS and ITAPS, the SLAC team has developed a systematic approach to determine shape deviations using measured frequencies and field values of dipole and monopole modes. New algorithms are under development to include additional measured external quality factors of dipole modes.

The shape determination is formulated as a weighted least squares minimization problem. The objective function composed of least squares differences between the measured and computed frequencies and field data of an accelerating cavity. The constraint is the Maxwell equations expressed in the frequency domain. Inversion variables are the shape deformations.

$$\begin{aligned} \underset{\mathbf{d}}{\text{minimize}} \quad \mathcal{J} &= \frac{1}{2}\alpha\,\Delta\lambda^T\Delta\lambda + \frac{1}{2}\beta\sum_{j=1}^{m}\left(\Delta E^j\right)^T\Delta E^j \\ \text{subject to} \quad & \mathbf{Ke} - \lambda\mathbf{Me} = \mathbf{0} \\ & \frac{1}{2}\mathbf{e}^T\mathbf{Me} = \frac{1}{2} \end{aligned}$$

where α and β are weighting constants, $\Delta\lambda$ and ΔE are misfits of frequencies and electric field, respectively.

A gradient based algorithm is used to solve the above optimization. The shape gradients are computed using a discrete adjoint approach and consistent with the discrete objective function. The non-linear problem is solved with a Gauss-Newton method with truncated singular value decomposition to deal with the ill-posedness. More details are described in [1].

Figure 1. The TESLA cavity for Internaltional Linear Collider.

The algorithm has been tested with synthetic problems [1]. Nonlinear convergence usually takes less than 20 iterations. Figure 2 shows the results of an inversion problem with measured data for a TESLA cavity shown in Figure 1. The blue line shows the misfit between the measured frequencies and those computed from the ideal cavity while the red line shows the inverted cavity misfit. The problem used 82 inversion variables, which are the unknown shape parameters including changes in cell radius, cell thickness, cell length, and iris radius.

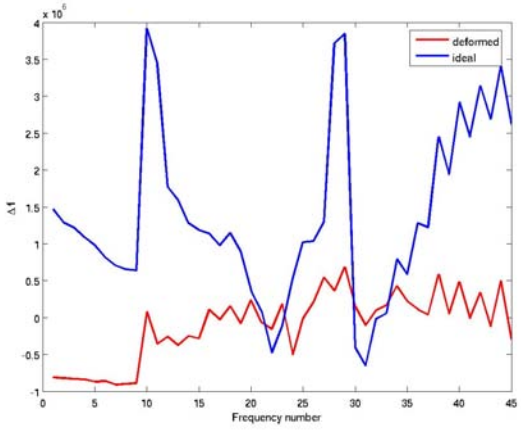
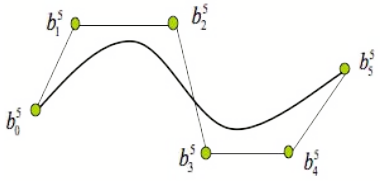


Figure 2. Frequency misfits for inverted and ideal cavities.

3. Element correction for meshes with complex geometries

The high-order finite element method used by SLAC for complicated accelerator structures requires the use of properly curved elements. The conventional unstructured mesh generators may create invalidly-curved elements at the boundaries. These invalid elements have negative determinants of the Jacobian in their closures, and may lead to inaccurate results for frequency domain analysis and divergences in time domain simulations.

A procedure, developed at RPI, applies Bezier polynomial to represent the mesh geometry and performs local mesh modification operations to incrementally correct the invalid elements. Bezier mesh geometry shapes are built on Bernstein polynomials [2] as shown in Figure 3,



$$x^q(\xi) = \sum_{i=0}^{q} B_i^q(\xi) b_i^q, \quad 0 \le \xi \le 1$$

$$\mathbf{x}(\xi_1, \xi_2, \xi_3) = \sum_{|i|=q} B_{|i|}(\xi_1, |xi_2, \xi_3) b_{|i|} = \frac{q!}{i!j!k!} \xi_1^i \xi_2^j \xi_3^k b_{|i|}$$

$$|i| = i + j + k = q$$

Figure 3. Bezier curve and Bezier mesh region geometric shape.

The mesh region validity is determined through the evaluation of the determinant of Jacobian described in the following [3]. Let *J* be the determinant of Jacobian of a curved Bezier element, which can expressed as

$$J = \left(\frac{\partial x}{\partial \xi_1} \times \frac{\partial x}{\partial \xi_2}\right) \cdot \frac{\partial x}{\partial \xi_3}$$

where $\frac{\partial x}{\partial \xi_1}$, $\frac{\partial x}{\partial \xi_2}$, and $\frac{\partial x}{\partial \xi_3}$ the three partial derivatives, which are also Bezier functions. Thus, *J* is also a Bezier polynomial of order *3(q-1)* written as,

$$J(\xi_1, \xi_2, \xi_3, \xi_4) = \sum_{|i|=3(q-1)} B_{|i|}(\xi_1, \xi_2, \xi_3, \xi_4) b'_{|i|}$$

Using the bounding property of the Bezier polynomials,

$$min(b'_{|i|}) \le J \le max(b'_{|i|})$$

Therefore, a curved element is valid if $min(b'_{|i|}) > 0$ that ensures the Jacobian of the element in its closure is always positive and is independent of the chosen integration scheme in the high-order finite element method.

The curved local mesh modifications are applied in a properly order that can effectively eliminate the invalid curved elements and make the resulting mesh valid [3]. The operations are built upon a set of operations including collapse, split, swap, and shape modifications [4]. Figure 4 showed the invalid tetrahedrons in yellow and its correct mesh inside an accelerator cavity.

The mesh correction procedure has been successfully applied to many meshes used in accelerator simulations. The corrected meshes not only make the time-domain simulations stable but also make the execution time up to 30% faster due to better conditioned matrices.

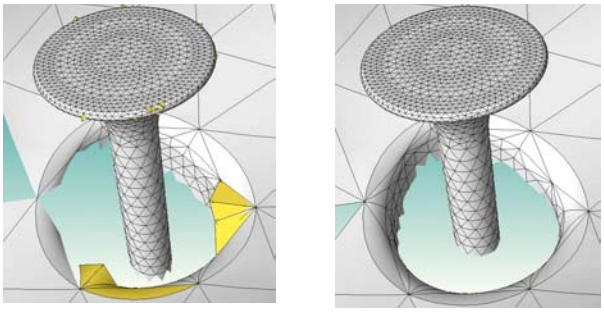


Figure 4. Invalid tetrahedrons and their corrections.

4. Memory reduction in linear solvers

Developing efficient and scalable linear solver is one of the keys to successful frequency-domain analysis of large accelerator structures using Omega3P. The use of sparse direct solvers is very effective in solving highly indefinite linear systems emerged from the accelerator simulations. On the other hand, its large memory usage is also a limiting factor for the simulation capability even on supercomputers. In particular, the unbalanced per-process memory usage from modern sparse direct solver packages severely limited the size of the problem that can be solved on the DOE's flagship computers. Various methods have been developed [6] to reduce the per-process memory usage at the expense of longer run time.

Method 1. The factor matrix and the stack for frontal matrices, the dominating items that consume a large amount memory in a multi-frontal sparse direct solver, are often stored in double precision in the double precision floating point computations. In this method, they are stored in single precision and the factor matrix is used as a preconditioner in the Krylov subspace methods. Because a single precision number occupies only 4 bytes while a double precision number has 8 bytes, the method halves the memory usage.

Method 2. In the complex linear systems emerged from Omega3P simulations, the complex matrix is very close to its real part. Namely, its imaginary part has a very-low rank and a small matrix norm. In this method, only the real part of the complex matrix is factorized and used as preconditioner in the Krylov subspace methods. This also halves the memory usage.

Method 3. This method combines Method 1 and 2. A real matrix is factorized in single precision and is used as preconditioner. Memory usage is about one quarter of that for a completed factorization of a double precision complex matrix.

Method 4. Reference [5] presented a hierarchical preconditioner based on the finite element order, which is suitable for solving real linear systems emerged from close cavity simulations. The idea has been extended to solve the complex linear systems emerged from open cavity simulations. In this method, only a small portion of a complex matrix corresponding to linear finite element basis functions is factorized. This significantly reduces memory usage.

Method 5. The method combines Method 1 and 4. A portion of a complex matrix is factorized in single precision.

Method 6. The method combines Method 2 and 4. The real part of a portion of a complex matrix is factorized.

Method 7. The method combines method 1, 2 and 4. The real part of a portion of a complex matrix is factorized in single precision.

To test the effectiveness of the above methods, computer experiments for solving a complex linear system with matrix size of 615114 and 23.7 million non-zeros were carried out. Table 1 summarized the results. With method 1 and 2, the per-process memory usage of a simulation cuts in half. With method 3, the memory usage cuts to about one quarter. With method 4, the memory usage cuts to about one tenth with a significantly longer run-time. Methods 5, 6, and 7 further reduced memory usage but the reducing ratio is not so large because of the relatively small testing matrix size. It should be noted that the reduction of memory usage is at the expense of the longer runtime.

Table 1. Various methods are tested for a complex linear system with matrix size of 615114 and 23.7 million non-zeros. MUMPS is a sparse direct solver. In all the other methods, a restarted generalized minimal residual (GMRES) with appropriate preconditioner was used. The third column is the maximal per-process memory usage. The last column is the time to compute a double-precision solution excluding factorization time. The numbers in the last column is more important than the factorization time because the factorization is performed once and the operation for computing a solution is performed hundreds of times for different right hand sides in our production simulations.

Method	Total memory(MB)	Per-process memory(MB)	Factorization Time (s)	Solution Time (s)
MUMPS	9862	782	41.3	1.03
1	5199	410	15.6	11.46
2	5457	429	9.25	114.1
3	3001	234	6.4	323.0
4	854	79	0.81	23.6
5	479	51	0.91	27.6
6	500	51	0.51	55.0
7	302	37	0.41	72.1

Advances in applied mathematics and computer science made it possible for SLAC scientists to simulate an 8-cavity ILC cryomodule shown in Figure 5, which is twice as large as the ILC superstructures simulated last year.

Figure 5. The electric field pattern of a trapped dipole mode inside an 8-cavity ILC cryomodule.

Acknowledgments

This work is supported by U.S. DOE ASCR, BES & HEP Divisions under contract DE-AC02-76SF00515. The work used resources of the National Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725, and the resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098.

References

[1] V. Akcelik, *et al.*, Shape Determination for Deformed Electromagnetic Cavities, J. of Comp. Phys., submitted.

[2] G. Farin, Curves and surfaces for computer aided geometric design, Academic Press, 1992.

[3] X. Luo, *et al.*, Automatic p-version mesh generation for curved domains, Engng. with Compters, v20, 273-285, 2004.

[4] X. Luo, An automatic adaptive directional variable p-versionmethod in 3-D curved domains, Ph.D thesis, Rensselaer Polytechnic Institute, 2005

[5] L. Lee, *et al.*, Solving large sparse linear systems in end-to-end accelerator structure simulations, in Proceedings of the 18th International Parallel and Distributed Processing Symposium, Santa Fe, New Mexico, April 26-30, 2004

[6] L. Lee, *et al.*, On reducing memory usage of the accelerator simulations using sparse direct solvers, SIAM Conference on Computational Science and Engineering, Costa Mesa, California, Feb. 19-23, 2007