A Parallel Poisson Solver Library for Accelerator Modeling Applications

Ji Qiang*[1], He Zhang[2], Jean-Luc Vay[1], Axel Huebl[1], David Grote[3], Kiran Sonnad [4], and Chad Mitchell[1]

[1]Lawrence Berkeley National Laboratory, Berkeley, CA, USA
[2]Thomas Jefferson National Accelerator Facility, Newport News, VA, USA
[3]Lawrence Livermore National Laboratory, Livermore, CA, USA
[4]Tech-X Corporation, Boulder, Co, USA

August 31, 2020

1 Introduction

Particle accelerators play a key role in high energy and nuclear physics studies, and many other applications. Design and operation of those sophisticated and expensive devices require extensively using multi-physics computer simulations. Those multi-physics computer models include space-charge effects, beam-beam effects, electron cloud effects, and other physical effects. In order to simulate the aforementioned effects self-consistently, one has to solve the Poisson equation at each step. A parallel, fast Poisson solver will be critical for the quick return in accelerator modeling applications.

In the accelerator community, a number of fast Poisson solvers were used and developed for accelerator modeling applications [1–24]. Subject to different boundary conditions, those Poisson solvers involve different numerical methods that solve the Poisson's Equation on a grid. For an open boundary condition, an FFT-based Green's function method was developed [3–10]. For a closed boundary condition with regular shape, a finite difference spectral method was used [11–15]. For a closed boundary condition with irregular shape, a multigrid finite difference method is often used [16–18, 25].

Besides solving the differential Poisson equation directly, the integral equation method provides a category of alternative solvers based on the integral format. These solvers convert the differential equation into an integral equation

*jqiang@lbl.gov

on the boundary. They do not need the grid that covers the whole domain and thus avoid any issues caused by the grid. The fast multipole method [19, 20], which has been implemented in a few accelerator simulation tools [21, 22], can be especially efficient with open boundaries. The integral equation method can also be applied to boundary value problems with any geometry. Finally, we note that some algorithms and codes [23, 24, 26] have been developed by computational mathematicians but have not been transplanted into the accelerator community.

2 Current and future challenges

So far, those Poisson solvers were developed on different computer platforms for accelerator modeling applications. Some of the implementations are serial while others take advantage of parallel architectures. They were also developed using different programming languages, e.g. Fortran, C++, Python or MatLab. There is also no uniform interface for accelerator modelers to use those Poisson solvers conveniently or interchangeably. Furthermore, the parallel Poisson solvers have often not been optimized for good efficiency and scalability on massive parallel computers, and very few have been ported to GPUs.

3 Advances needed to meet challenges

In order to meet the above challenges, we propose to develop a fast, portable, parallel Poisson solver library, which would improve the usage of those fast Poisson solvers and benefit the accelerator modeling applications. Modern programming practices and tools can be leveraged to develop the library that will be portable across CPUs and GPUs, and with standardized interfaces that make it easy for accelerator modelers to use in their applications, toolkits and ecosystem [27, 28]. Multiple levels of parallelism (on-node and multi-nodes) will be supported, with special attention to efficiency and scalability. The library will also include a detailed and up-to-date user documentation, as well as automated test suites and well-benchmarked examples.

References

[1] Roger W Hockney and James W Eastwood. *Computer simulation using particles*. New York: McGraw-Hill Book Company, 1985.

[2] William T Vetterling and William H Press. *Numerical recipes in Fortran: the art of scientific computing*, volume 1. Cambridge University Press, 1992.

[3] Kazuhito Ohmi. Simulation of beam-beam effects in a circular e+ e- collider. *Physical Review E*, 62(5):7287, 2000.

[4] R. D. Ryne. Icfa beam dynamics mini workshop on space charge simulation, trinity college, 2003.

[5] Ji Qiang, Miguel A Furman, and Robert D Ryne. A parallel particle-in-cell model for beam–beam interaction in high energy ring colliders. *Journal of Computational Physics*, 198(1):278–294, 2004.

[6] Ji Qiang, Steve Lidia, Robert D Ryne, and Cecile Limborg-Deprey. Three-dimensional quasistatic model for high brightness beam dynamics simulation. *Physical Review Special Topics-Accelerators and Beams*, 9(4):044204, 2006.

[7] Ji Qiang, Steve Lidia, Robert D Ryne, and Cecile Limborg-Deprey. Erratum: Three-dimensional quasistatic model for high brightness beam dynamics simulation [phys. rev. st accel. beams 9, 044204 (2006)]. *Physical Review Special Topics-Accelerators and Beams*, 10(12):129901, 2007.

[8] Valentin Ivanov. Green's function technique in forming intensive beams. *International Journal of Modern Physics A*, 24(05):869–878, 2009.

[9] Ji Qiang, Miguel A Furman, and Robert D Ryne. Strong-strong beam-beam simulation using a green function approach. *Physical Review Special Topics-Accelerators and Beams*, 5(10):104402, 2002.

[10] Ji Qiang, Miguel A Furman, Robert D Ryne, Wolfram Fischer, and Kazuhito Ohmi. Recent advances of strong–strong beam–beam simulation. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 558(1):351–355, 2006.

[11] Ji Qiang and Robert D Ryne. Parallel 3d poisson solver for a charged beam in a conducting pipe. *Computer physics communications*, 138(1):18–28, 2001.

[12] Ji Qiang and Robert L Gluckstern. Three-dimensional poisson solver for a charged beam with large aspect ratio in a conducting pipe. *Computer physics communications*, 160(2):120–128, 2004.

[13] Ji Qiang. Efficient three-dimensional poisson solvers in open rectangular conducting pipe. *Computer Physics Communications*, 203:122–127, 2016.

[14] Ji Qiang. A fast parallel 3d poisson solver with longitudinal periodic and transverse open boundary conditions for space-charge simulations. *Computer Physics Communications*, 219:255–260, 2017.

[15] Ji Qiang. Fast 3d poisson solvers in elliptical conducting pipe for space-charge simulation. *Physical Review Accelerators and Beams*, 22(10):104601, 2019.

[16] Wolfgang Hackbusch. *Multi-grid methods and applications.* New York: Springer-Verlag, 1985.

[17] Pieterg Wesseling. *Introduction To Multigrid Methods.* Chichester: John Wiley & Sons, 1992.

[18] Ji Qiang, D Todd, and Daniela Leitner. A 3d model for ion beam formation and transport simulation. *Computer physics communications*, 175(6):416–423, 2006.

[19] J Carrier, Leslie Greengard, and Vladimir Rokhlin. A fast adaptive multipole algorithm for particle simulations. *SIAM journal on scientific and statistical computing*, 9(4):669–686, 1988.

[20] Hongwei Cheng, Leslie Greengard, and Vladimir Rokhlin. A fast adaptive multipole algorithm in three dimensions. *Journal of computational physics*, 155(2):468–498, 1999.

[21] He Zhang and Martin Berz. The fast multipole method in the differential algebra framework. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 645(1):338–344, 2011.

[22] Steffen Schmid, Erion Gjonaj, and Herbert de Gersem. 3d space charge tracking using fast multipole methods. *Verhandlungen der Deutschen Physikalischen Gesellschaft*, 2018.

[23] YJ Liu and N Nishimura. The fast multipole boundary element method for potential problems: a tutorial. *Engineering Analysis with Boundary Elements*, 30(5):371–381, 2006.

[24] Andreas Klöckner, Alexander Barnett, Leslie Greengard, and Michael O'Neil. Quadrature by expansion: A new method for the evaluation of layer potentials. *Journal of Computational Physics*, 252:332–349, 2013.

[25] Warp, http://warp.lbl.gov.

[26] Matt Wala and Andreas Klöckner. A fast algorithm for quadrature by expansion in three dimensions. *Journal of Computational Physics*, 388:655–689, 2019.

[27] David Sagan et al. Beam dynamics toolkit. *Snowmass21 LOI*, 2020.

[28] Jean-Luc Vay et al. A community ecosystem of toolkits for multiphysics particle accelerator modeling and design. *Snowmass21 LOI*, 2020.