ISBN: 978-3-95450-168-7 doi:10.18429/JACoW-IPAC2015-MOPWA003

OPTIMAL GENERALIZED FINITE DIFFERENCE SOLUTION TO THE PARTICLE-IN-CELL PROBLEM

X. Wang, Stony Brook University, Stony Brook, NY 11794, USA
R. Samulyak, Stony Brook University, Stony Brook, NY 11794, USA and BNL, NY 11973, USA
X. Jiao, Stony Brook University, Stony Brook, NY 11794, USA
K. Yu, Stony Brook University, Stony Brook, NY 11794, USA

Abstract

A new adaptive Particle-in-Cloud (AP-Cloud) method for obtaining optimal numerical solutions to the Vlasov-Poisson equation has been proposed. The traditional particle-in-cell (PIC) method, commonly used for solving this problem, is not optimal in terms of the balance of errors of the differential operator discretization and source integration; it is also inaccurate when the particle distribution is highly non-uniform. Our method replaces the Cartesian grid in the traditional PIC with adaptive computational nodes or particles, to which the charges from the physical macroparticles are assigned by a weighted least-square approximations. The partial differential equation is then discretized using a generalized finite difference (GFD) method and solved with fast linear solvers. The density of computational particles is chosen adaptively, so that the error from GFD and that from the source integration are balanced and the total error is approximately minimized. The method is independent of geometrical shape of computational domains and free of artificial parameters. Results of verification tests using electrostatic problems of particle beams with halo and comparison of accuracy and solution time of the AP-Cloud method with the traditional PIC are presented.

ERROR ANALYSIS OF TRADITIONAL PIC METHOD

Particle-in-cell [1] (PIC) is the traditional method for solving both the Vlasov-Poisson and Vlasov-Maxwell problems. In this work, we focus on the Vlasov-Poisson equation using an example of the electrostatic space charge problem for particle beams.

In PIC, the charge density at grid point $\rho(x^j)$ is estimated from the distribution of particles by interpolating particle charges to mesh nodes. Then the Poisson equation

$$\Delta\varphi = \rho \qquad (1)$$

subject to a Dirichlet or Neumann boundary condition is discretized on the mesh. Performing error analysis of the PIC method, we can show that the total error is

$$O\left(\sqrt{\frac{\rho(x^i)}{Nh^D}} + \rho(x^i)h^2\right),$$

where N is the number of physical macroparticles, h is the cell size, and D is the space dimension. The error is minimized if

$$h = O\left(\frac{1}{N\rho(x^j)}\right)^{\frac{1}{4+D}},$$

which is impossible if a uniform mesh is used for a highly nonuniform particle distribution.

AP-CLOUD METHOD

Adaptive Particle-in-Cloud (AP-Cloud) method can be viewed as a meshless and adaptive version of PIC. We use computational particles instead of Cartesian grid, the distribution of which is derived using an error balance criterion. Instead of the finite difference discretization of the Laplace operator, we use the framework of weighted least squares approximation, also known as the generalized finite-difference (GFD) [2]. The framework includes interpolation, least squares approximation, and numerical differentiation on a stencil in the form of cloud of computational particles in a neighborhood of the point of interest. It is used for the charge assignment scheme, numerical differentiation, and interpolation of solutions.

The Particle-in-Cloud method operates as follows:

- Given a distribution of physical macro-particles, optimally select a subset of computational nodes (particles) from this distribution by constructing an octree and applying the error balance criterion
$$h = O\left(\frac{1}{N\rho(x^j)}\right)^{\frac{1}{2k+D-2}}$$
where h is the local averaged distance between computational macro-particles and k is the order of interpolation polynomial in the GFD method.
- Place computational particles on the boundary
- Enforce the 2:1 balance of inter-particle distances in the case of extreme density changes. The 2:1 balance requires that the difference between the levels of refinement of two neighbors is at most one, improving the smoothness in the placement of computational particles.
- Assign physical states to computational nodes and approximate differential operators in the location of computational nodes using GFD.
- Solve the corresponding linear system using a

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2015). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

ISBN: 978-3-95450-168-7 doi:10.18429/JACoW-IPAC2015-MOPWA003

fast parallel solver.

- Calculate the solution gradient on computational particles using the same GFD stencils.
- Interpolate gradients back to the location of macroparticles using Taylor expansion.

Another important feature and advantage of the Particle-in-Cloud method is its ability to solve problems in irregular and geometrically complex domains.

NUMERICAL VERIFICATION TESTS

We have performed verification of the Particle-in-Cloud method using problems with highly non-uniform distributions of particles typical for accelerator beams with halos. We study a high-intensity, small-sigma particle beam is surrounded by a larger radius halo containing from 3 to 6 orders of magnitude smaller number of particles (see Figure 1).

2D Particle Beam with Halo

Consider the Poisson equation (1) in a unit square and assume that the particle density is given by the following function

$$\rho = a_1\left[\exp\left(-\frac{\|\mathbf{x}\|^2}{\sigma_1^2}\right) + a_2\exp\left(-\frac{\|\mathbf{x}\|^2}{\sigma_2^2}\right)\right] \qquad (2)$$

where σ_1=0.002, σ_2=0.3, a_2=10^{-5}, and a_1=396.1 is the normalization parameters to ensure $\int_\Omega \rho(\mathbf{x})\,d\mathbf{x} = 1$. To obtain a benchmark solution, we embed the unit square in a larger disk with zero Dirichlet boundary condition, and solve the corresponding Poisson problem very accurately in 1D using the radial coordinate. The boundary conditions for the 2D problems are obtained by using the 1D solution with proper radial coordinates corresponding to the unit square. In the 2D domain, physical macroparticles are randomly generated with the density (2), and the problem is solved using the traditional PIC and the AP-Cloud method. This procedure was designed to enable calculations with the traditional PIC on a square mesh; the AP-Cloud method is independent of the geometry.

Figure 1 shows the distribution of physical macroparticles, colored according to numerical solution values for the potential and Figure 2 shows the placement of computational particles. Comparison of errors of the traditional PIC and the AP-Cloud is presented in Figure 3. For similar computational time, the Particle-in-Cloud method achieves smaller errors of the solution gradient by the factor of 30 – 50 compared to PIC.

Comparable results were obtained for the extension of problem (1) – (2) to 3D. The computational test used 10^6 macroparticles. Figure 4 shows the distribution of 4067 computational particles, selected by the particle placement algorithm.

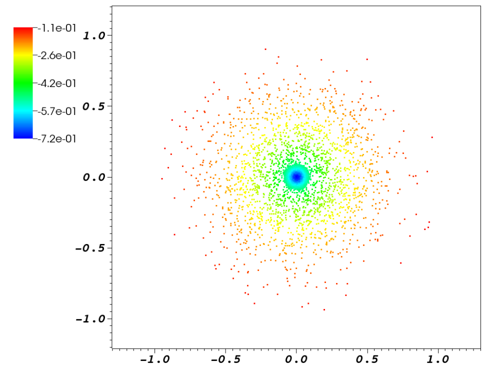

Figure 1: Typical distribution macroparticles in transverse cross section of accelerator beam with halo. Color represents numerical solution values for the potential.

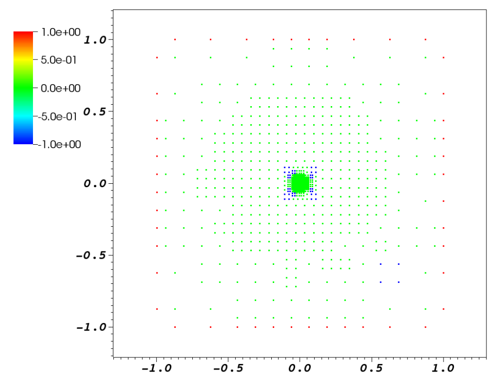

Figure 2: Typical particle distribution of computational particles in the beam halo problem.

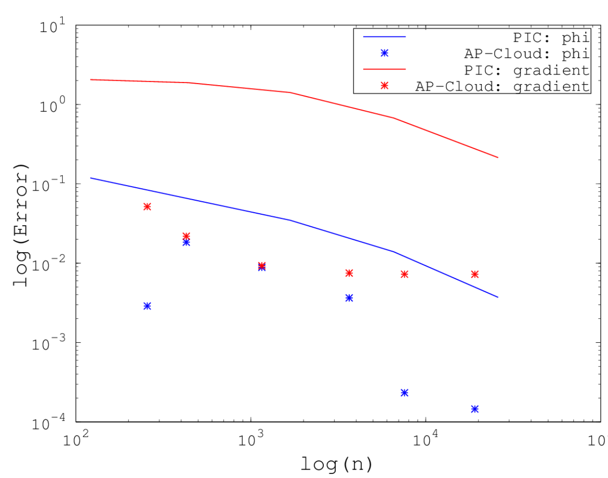

Figure 3: Errors of PIC and AP-Cloud method as a function of number of grid points or computational particles (nodes).

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2015). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

ISBN: 978-3-95450-168-7 doi:10.18429/JACoW-IPAC2015-MOPWA003

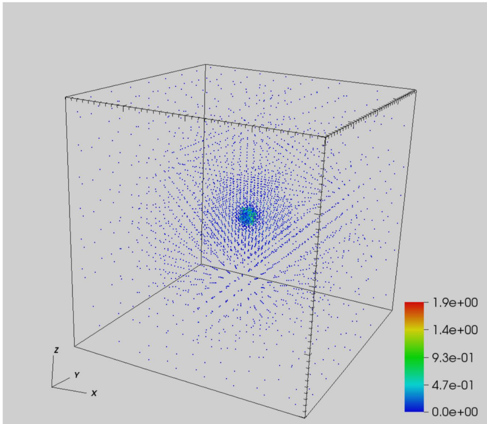

Figure 4: Distribution of 4067 computational particles with values of electric field error.

Self-force Effect with Single Particle

Vlasov-Poisson problems with highly non-uniform distributions of matter can be solved using the adaptive mesh refinement technique in PIC [3.4]. It is however well known that AMR-PIC introduces significant artifacts in the form of artificial image particles across boundaries between coarse and fine meshes. These images introduce spurious forces that may potentially alter the particle motion to an unacceptable level [3]. Methods for the mitigation of the spurious forces have been designed [4]. The traditional PIC on a uniform mesh is free of such artifacts.

While the convergence of Adaptive Particle-in-Cloud solutions to benchmark (highly resolved 1D) solutions already indicates the absence of artifacts, we have performed additional tests similar to the one in [3], specially designed to investigate the presence of artificial images. In the AMR-PIC case, the problems involves the motion of a single particle across the coarse – fine mesh interface. For the Particle-in-Cloud method, we studied the motion of a single test particle represented by a moving cloud of computational particles with refined distances towards the test particle. The test particle contained a smooth, sharp, Gaussian-type charge distribution to satisfy requirements of the GFD method.

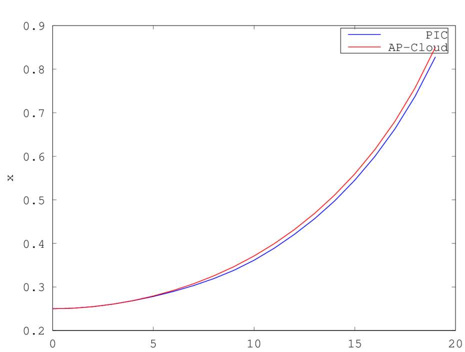

Figure 5: Motion of a single test particle obtained with PIC and AP-Cloud methods demonstrating the absence of artifacts in the AP-Cloud method.

The motion of a single test particle obtained with PIC and Particle-in-Cloud methods is shown in Figure 5. The test provides an additional assurance that artificial images are not present in the AP-Cloud method.

CONCLUSIONS AND FUTURE WORK

We have developed an Adaptive Particle-in-Cloud (AP-Cloud) method that replaces the Cartesian grid in the traditional PIC with adaptive computational nodes. Adaptive particle placement balances the errors of the differential operator discretization and the source computation (equivalent to the error of the Monte Carlo integration) to minimize the total error.

AP-Cloud uses GFD based on weighted least squares (WLS) approximations on a stencil of irregularly placed nodes. The framework includes interpolation, least squares approximation, and numerical differentiation capable of high order convergence.

The Particle-in- Cloud method has significant advantages over the traditional PIC for non-uniform distributions of particles and complex boundaries. It achieves 30 - 50 times better accuracy in the gradient of the potential compared to the traditional PIC for the problem of particle beam with halo. The method is independent of the geometric shape of the computational domain, and can achieve highly accurate solutions in geometrically complex domains. Specially designed tests showed that the AP-Cloud method is free of artificial images and spurious forces typical for the first versions of AMR-PIC (without special mitigation techniques).

Future work will focus on higher convergence rates of the method, timing optimization, and scalability. Future application will also include cosmological simulations of dark matter that are characterized by the formation of halos.

ACKNOWLEDGMENT

This research was supported in part by BNL LDRD grant.

REFERENCES

[1] R.W. Hockney, J.W. Eastwood, Computer simulation using particles, (CRC Press, 1988), 21.
[2] J.J. Benito, F. Urena, L. Gavete, Infuence of several factors in the generalized finite difference method, Applied Math. Modeling, 25 (2001), 1039-1053.
[3] Vay, J-L., et al., Mesh refinement for particle-in-cell plasma simulations: applications to and benefits for heavy ion fusion, Laser and Particle Beams, 20.04 (2002): 569-575.
[4] P. Colella, and P.C. Norgaard. Controlling self-force errors at refinement boundaries for AMR-PIC, J. Comput. Phys. 229.4 (2010): 947-957.

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2015). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.