

# SOLVING BY PARALLEL COMPUTATION THE POISSON PROBLEM FOR HIGH INTENSITY BEAMS IN CIRCULAR ACCELERATORS\*

A.U. Luccio, N.L. D'Imperio, R.Samulyak, J. Beebe-Wang, BNL, Upton, NY11973-5000, USA

## Abstract

Simulation of high intensity accelerators leads to the solution of the Poisson Equation, to calculate space charge forces in the presence of acceleration chamber walls. We reduced the problem to “two-and-a-half” dimensions for long particle bunches, characteristic of large circular accelerators, and applied the results to the tracking code *Orbit*.

## 1 THE SPACE CHARGE PROBLEM

In PIC tracking of an accelerator, a “herd” of randomly generated macro particles are pushed through a lattice represented by a sequence of maps[3]. Once all the macros have reached certain locations, SC (space charge) nodes, their density  $\rho$  is calculated by binning to a grid, and the potential  $\Phi$  is found by solving the Poisson Equation with boundary (wall) conditions

$$\nabla^2 \Phi = -\frac{1}{\epsilon_0} \rho, \quad \rho_{wall}(x, y, z) = g(x, y, z) \quad (1)$$

From the potential  $\Phi$ , SC force components (with a coefficient to account for the balance between electrostatic and magnetic action), are calculated by derivation and applied to each macro as transverse angle kicks

$$\Delta \vec{p} = \int \vec{F} dt \quad (2)$$

In a ring with long bunches we may uncouple the transverse motion from the longitudinal

$$\rho(x, y, z) = \rho_z(z) \rho_u(x, y) \quad (3)$$

and solve the Poisson problem in parallel in many longitudinal beam slices, where macros are subject to different SC kicks, due to different transverse beam aspect ratios.

Since lattice map sequence controls the propagation, and the Poisson problem must be solved with all the macros evaluated at the same time, when the herd reach a SC node, each macro is longitudinally moved with respect to some reference (synchronous particle), using lattice transfer maps. Fig.1 show a beam bunch thus “frozen” at a given time in a simple FODO channel for SC calculation.

Limitations of this procedure are that maps used to put macros in their appropriate place are for the bare lattice (no extra focusing and tune shift due to SC forces), and that longitudinal forces between slices are disregarded.

A better approximation is obtained by decomposing also the potential  $\Phi$  in a longitudinal and a transverse part

$$\Phi(x, y, z) = \Phi_z(z) \Phi_u(x, y) \quad (4)$$

\* Work sponsored by the U.S. Department of Energy

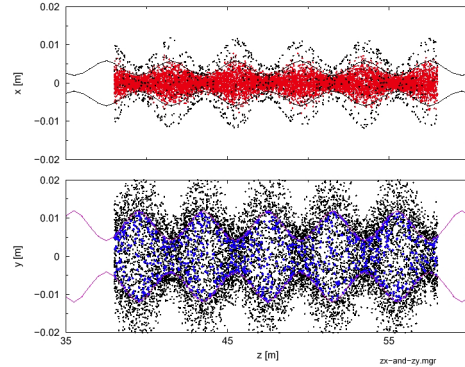


Figure 1: Frozen matched beam. Envelope is also shown.

In Cartesian coordinates Poisson becomes

$$\Phi_z \nabla_{\perp}^2 \Phi_u + \Phi_u \frac{\partial^2 \Phi_z}{\partial z^2} = -\frac{1}{\epsilon_0} \rho_z \rho_u \quad (5)$$

With both  $\rho_z(z)$  and  $\Phi_z(z)$  piece-wise constant, equate  $z$  and  $(x, y)$  functions on both sides of Eq.(5), to obtain

$$\Phi_z = \rho_z, \quad \nabla_{\perp}^2 \Phi_u = -\frac{1}{\epsilon_0} \rho_{\perp}(x, y). \quad (6)$$

Then, an approximate solution is obtained by solving for  $\Phi_u$  in the transverse space using  $\rho_u$ , and then multiply the result by the constant  $\Phi_z$

$$\Phi^{(0)}(x, y, z) = \Phi_z \Phi_u(x, y). \quad (7)$$

For a better solution, use a perturbative method. With  $\Phi_z^{(1)}(z) = \Phi_z + \phi(z)$  ( $\phi$  small) in Eq.5, find, after canceling out the lowest order terms

$$\frac{\partial^2 \phi(z)}{\partial z^2} + \omega^2 \phi(z) = 0, \quad \omega^2 = -\frac{\rho_u}{\epsilon_0 \Phi_u} \quad (8)$$

Assuming in the center of the beam:  $\phi(0) = 0$ , obtain a solution

$$\Phi_z^{(1)}(z) = \Phi_z + \frac{1}{\omega} \frac{\partial \Phi_z}{\partial z}(0) \cos(\omega z) \quad (9)$$

Note: (1) The frequency  $\omega$  is a (weak) function of  $(x, y)$ . i.e. transverse charge density and transverse potential have similar shape. (2) The derivative of the longitudinal potential in the center of the slice in Eq.(9) is approximately proportional to the longitudinal variation of current in the beam at that location, in agreement with the basic result of the impedance model.

## 2 POISSON SOLVERS

### 2.1 Integral Poisson Solvers

The integral formulation of Poisson equation is

$$\phi(P) = C \rho_z \int \frac{\rho_u(Q)}{|P-Q|} dQ \quad (10)$$

with  $P$  a field point,  $Q$  a source point, and  $C$  is a perveance coefficient including the factor  $\frac{1}{\gamma^2}$ .

A direct integration of Eq.10 (Brute Force, or BF) gives a very transparent solution, with limitation: (1) arbitrary treatment of poles arising from field points accidentally coincident with source points, (2) difficulties to include images on walls in the calculation, (3) length of execution.

Eq.(10) can be solved by convolution, by first performing a FFT of the Green function  $G(u) = \frac{1}{r}$  and of the charge density

$$\begin{pmatrix} \tilde{G}(\omega) \\ \tilde{\rho}_\perp(\omega) \end{pmatrix} = FFT \begin{pmatrix} G(u) \\ \rho_\perp(u) \end{pmatrix}. \quad (11)$$

and then find the potential by the anti-FFT of the convolution

$$\Phi(r) = C\rho_\parallel FFT^{-1} \left( \tilde{G}(\omega) * \tilde{\rho}_\perp(\omega) \right). \quad (12)$$

FFT needs a grid twice the size of the beam to avoid aliases.

## 2.2 Differential Poisson Solvers

The differential Poisson Eq.(1) can be solved by LU decomposition, by discretization on an  $N \times N$  grid

$$-4\pi\rho_{ij} = \sum_{kl} \mathcal{L}_{ij}^{kl} \Phi_{kl}, \quad \Phi(P) = -\frac{1}{4\pi} \mathcal{L}^{-1} \rho(Q) \quad (13)$$

In Cartesian coordinates, the Laplacian is written as

$$\mathcal{L}_{ij}^{kl} = -4\delta_i^k \delta_j^l + \delta_{i+1}^k \delta_j^l + \delta_{i-1}^k \delta_j^l + \delta_i^k \delta_{j+1}^l + \delta_i^k \delta_{j-1}^l. \quad (14)$$

$\mathcal{L}$  is a very large  $N^2 \times N^2$  band-sparse matrix. In general, its inverse is dense.

Eq. 13 can also be effectively solved by iteration. From

$$\rho_{i,j} = \frac{\Phi_{i-1,j} + \Phi_{i,j+1} + \Phi_{i+1,j} + \Phi_{i,j-1} - 4\Phi_{i,j}}{h^2} \quad (15)$$

it is, at iteration  $k$

$$\Phi_{i,j}^{k+1} = \frac{1}{4} \left( \Phi_{i-1,j}^k + \Phi_{i,j+1}^k + \Phi_{i+1,j}^k + \Phi_{i,j-1}^k - \rho_{i,j} \right). \quad (16)$$

## 2.3 Walls

The effect of walls is classically treated by using impedances  $Z_n$  [4], calculated as a function of frequency mode. Voltage kicks are applied to The FFT components of the current. Another approach with an electrical circuit analog can find how the beam is coupled to walls via E-M fields[5].

The Poisson Eq. allows one to calculate walls by adding to the true charges in the beam, image charges and currents on the walls. Images need to be known in advance for integral Poisson solvers. In differential solvers, knowledge of wall potential suffices, and images can be subsequently calculated. This suggests the interesting possibility that, in turn, Poisson solvers can be used to calculate impedances in odd wall geometries.

We must extend the grid to the walls. A Cartesian grid is often not suitable, since it will end up in a bad density of points at the wall, essentially ignoring corners. A cylindrical or an elliptical transverse grid, Fig.2, or more generally an adaptive grid is better fit to treat walls.

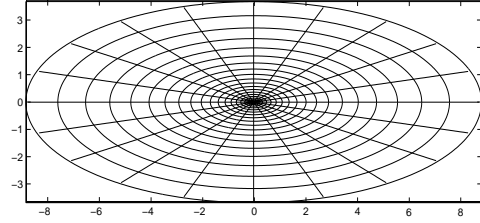


Figure 2: Elliptic mesh. Equal cell areas at a given radius

The Laplacian in the elliptic coordinate system is

$$\left( 1 + \frac{1}{\gamma^2} - \alpha^2 \right) \frac{\partial^2}{\partial r^2} - \frac{\beta}{r} \frac{\partial^2}{\partial r \partial \theta} + \frac{\alpha^2}{r^2} \frac{\partial^2}{\partial \theta^2} + \frac{\beta}{r^2} \frac{\partial}{\partial \theta} + \frac{\alpha^2}{r} \frac{\partial}{\partial r}$$

where the azimuthal coordinate  $\theta$  is the parameter along the grid lines,  $\gamma$  is the ratio of the ellipse axes,

$$\alpha^2 = \sin^2 \theta + \frac{1}{\gamma^2} \cos^2 \theta, \quad \beta = 2 \left( 1 - \frac{1}{\gamma^2} \right) \sin \theta \cos \theta.$$

The discrete 9-point Laplacian for the elliptic grid with a radial grid refinement ratio  $dr_{i+1}/dr_i = k$  is

$$\nabla^2 \Phi_{i,j} = A\Phi_{i+1,j} + B\Phi_{i-1,j} + C\Phi_{i,j+1} + D\Phi_{i,j-1} + E\Phi_{i,j} \\ + F\Phi_{i+1,j+1} + G\Phi_{i+1,j-1} + H\Phi_{i-1,j+1} + I\Phi_{i-1,j-1}$$

where the coefficients are functions of  $k, \alpha, \beta, \theta, dr, d\theta$ .

## 3 PARALLEL COMPUTING

To push a large herd of many representative macro particles around the lattice of an accelerator under SC forces we use parallel computers with many nodes, each consisting of a processor and associated memory. Our paradigm of choice was MPI (Message Passing Interface)[6].

### 3.1 Parallel Tracking

In rings particles may be tracked for many turns, from  $10^3$  to  $10^6$ . A typical ring lattice may contain hundreds to thousands transfer maps and SC nodes per turn.

When self forces are negligible, tracking is an ‘‘embarrassing parallel’’ problem, where each processor runs the same code on a subset of macro particles. Space charge dominated beams lead to much more complex parallel computing issues. In this case, The Poisson problem is dealt with following one of two possible strategies:(1) each processor pushes a sub-herd of particles residing anywhere along the beam, and (2) each processor pushes macros belonging to a given longitudinal slice, as described in Sec.1. The first strategy deals with the beam as a whole, however with much message passing, since each processor must

Table 1: Specifications for the BNL Galaxy Cluster[1].

Nodes:	2 CPU's	512 MB RAM
CPU specs:	Intel Pentium III	
500 MHz	512 kB L2 Cache	
Interconnection:	Cisco Switch	100 MB Ethernet
File System:	NFS	
MPI:	mpich	1.2 compliant

share the position of each macro at an SC node. The second presents a problem of ghosts, i.e. macros that because of longitudinal motion within the beam migrate between contiguous slices. MPI contains tools to address the problem of ghosts.

At a SC node, we solve Poisson Eq. The FFT method is straightforward in parallel: since the integrals are linear operators. The BF is not parallel with regards of number of macros involved. The 4-fold size of the nested loops to calculate the integral depends only on the grid size, the sums just being applied at each processor to smaller numbers, i.e. the individual partial charge densities.

For parallel LU, a set of  $\mathcal{L}$  matrices (one for each wall configuration) and their inverse needs to be calculated only once, at the beginning of the run. What has to be done at a space charge node is the matrix multiplication in parallel, by assigning to each processor a band of rows of the inverted matrix and multiply this band times all the  $\rho$ 's to find at each processor the potential or force for a region of the grid, as shown in Fig.3.

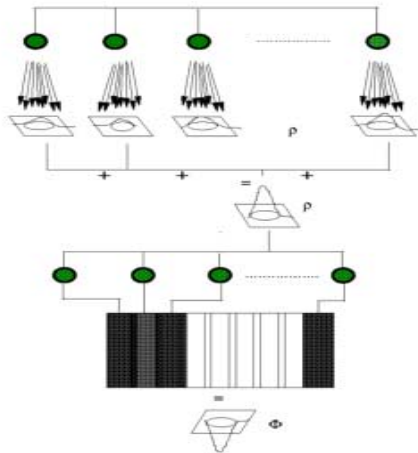


Figure 3: LU Parallel Strategy

### 3.2 Examples

We compared the speed of Poisson solvers (LU, FFT and BruteForce) for a herd of  $3.6 \cdot 10^6$  macros ( $10^5$  per process), using the Brookhaven Galaxy cluster described in Table 3.2 with 77 processors. The results are shown in Table 3.2. The  $x$ -force field is shown in Fig.4.

As a second application, we implemented a MPI version of the tracking code *Orbit*[2] with FFT Poisson solver and

Table 2: Results of Poisson solvers. Two grid sizes

	BF/FFT/LU	
Grid:	33/65/33	65/129/65
	Elapsed Time [s]	
Lu Solver	0.0279	0.1407
FFT	0.0538	0.2558
BruteForce	0.1343	2.1300

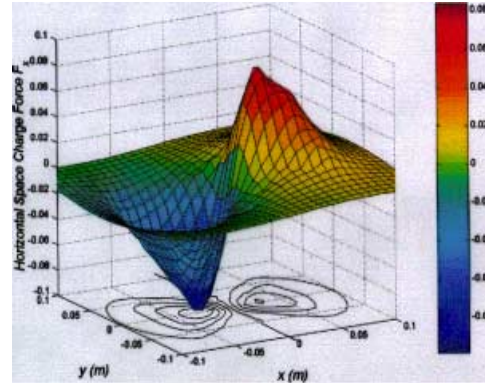


Figure 4: Horizontal component of the Force.

run it in parallel on the Galaxy. Timing results are given in Table 3. A number of macros exceeding  $25 \cdot 10^6$  is not at the present time possible due to memory limits. Linear performance scaling was observed with 32 processors.

## 4 REFERENCES

- [1] CDIC, The Center for Data Intensive Computing (J. Glimm, Director), BNL and SUNY-Stony Brook, [www.bnl.gov/cdic](http://www.bnl.gov/cdic)
- [2] J.D. Galambos, J.A. Holmes, D.K. Olsen, A. Luccio, J. Beebe-Wang, 'Orbit User Manual Version 1.10', SNS/ORNL/AP Technical Note 011, Rev.1, 1999 [www.sns.gov/APGroup/Codes/Codes.html](http://www.sns.gov/APGroup/Codes/Codes.html)
- [3] R.W. Hockney and J.W. Eastwood, 'Computer Simulation Using Particles', Adam Hilger, IOP Pub, New York, 1988
- [4] A.W. Chao, 'Physics of Collective Beam Instabilities in High Energy Accelerators', Wiley, NY 1993.
- [5] D.A. Callahan Miller, 'Simulation of Longitudinal Beam Dynamics of Space-Charge Dominated Beams for Heavy Ion Fusion', PHD Thesis, University of California, Davis, UCRL-LR-119364, December 1994
- [6] W. Gropp, E. Lusk, A. Skjellum, 'Using MPI', MIT Press, 1998

Table 3: *Orbit* timing (wall clock) on the Galaxy. 1 Turn.

Nodes	Macros/Turn per Node	With SC	No SC
		time sec	time sec
2	$1.6 \cdot 10^6$	1934	818
9	$0.2 \cdot 10^6$	253	85
17	$0.1 \cdot 10^6$	142	42
33	$0.05 \cdot 10^6$	88	23