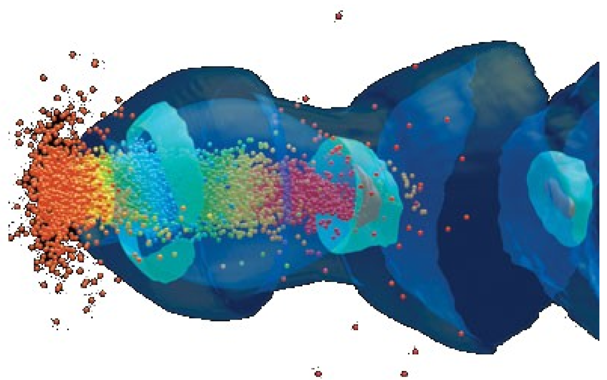
A Fast Parallel Poisson Solver for Beam Dynamics

A. Adelmann[1], P. Arbenz[2], Y. Ineichen[2]

[1] Paul Scherrer Institut, PSI — [2] ETH Zuerich

Space-Charge in the Electrostatic Approximation



- Electric field caused by Coulomb repulsion
- Magnetic field arising from moving particles
- Lorentz transformation to obtain a electrostatic problem
- Arising Poisson equation: $\Omega \subset \mathbb{R}^3$

$$\nabla^2 \phi = -\frac{\rho}{\epsilon_0}$$

- The electric field from electrostatic potential:

$$E = -\nabla \phi$$

- The magnetic field is calculated from the electric field by the inverse Lorentz transformation

PIC

- Interpolate particle charges to a grid
- Solve Poisson equation in Lorentz frame
- *O(n log n)*
- CIC interpolation with **IPPL**[1] providing: parallel fields, particle representation, operators on fields

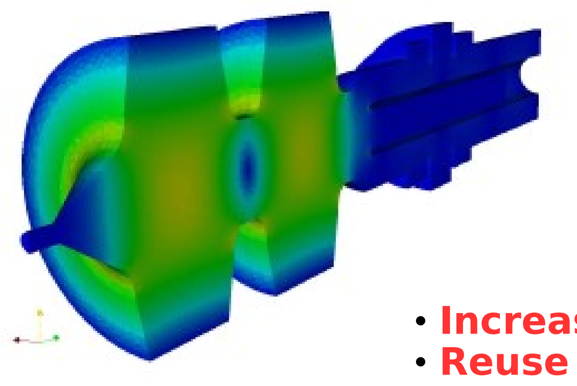
Direct Solver (FFT)

- Convolution with Green's function
- Integrated Green's function
- Shifted Green's function for image charge at cathode

- Rectangular domain with free-space boundary conditions

- **Fast solver**
- **State of the art method**
- **Implemented and tested in late 2007**

Iterative Solver (Preconditioned CG)



- AMG preconditioned CG solver
- Anisotropic grid in z-direction

- Elliptic and arbitrary bounded domains with Dirichlet (PEC) and Robin boundary conditions

- **Increase accuracy of simulation**
- **Reuse solution of previous time-step**

AMG

- Decoupled aggregation
- Polynomial smoother
- LU based direct solver
- *O(n)* algorithm
- Implemented with **Trilinos**[2] packages: **ML**, **Amesos**, **AztecOO**, **Epetra**
- Implementation reuses solution of previous time-step
- Added an interface between **IPPL** and **Epetra**
- Reusing aggregation hierarchy

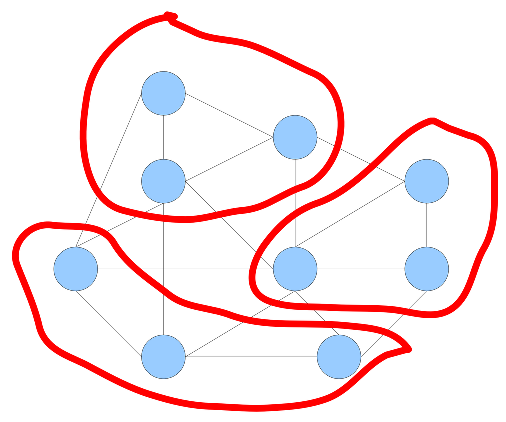
Multigrid V-Cycle Algorithm

```
procedure MultiGridSolve(A_l,b_l,x_l,l)
  if l = maxLevel-1 then
    DirectSolve A_l x_l = b_l
  else
    x_l ← S_l^pre(A_l,b_l,0)
    r_l ← b_l-A_l x_l {calculate residual}
    b_l+1 ← R_l r_l {restriction}
    v_l+1 ← 0
    MultiGridSolve(A_l+1,b_l+1,v_l+1,l+1)
    x_l ← x_l + P_l v_l+1 {coarse grid correction}
    x_l ← S_l^post(A_l, b_l, x_l)
  end if
end procedure
```

Aggregation

- Convert discretization matrix into graph
- Assign each vertex to one aggregate
- Tentetative prolongator operator matrix is formed:

$$p_{i,j} = \begin{cases} 1 & \text{if } i^{th} \text{ vertex} \in j^{th} \text{ aggregate} \\ 0 & \text{otherwise} \end{cases}$$



- Improve robustness by smoothing the tentative prolongator operator

BC

- Various discretization approaches *O(h)* and *O(h²)*
- Depending on boundary approximation: non-symmetric discretization matrix
- Workflow for arbitrary boundary geometries from **STEP** geometries to **H5fed**[3] files
- Efficient intersection of grid-lines with triangulated surface mesh[4]

Boundary Problem

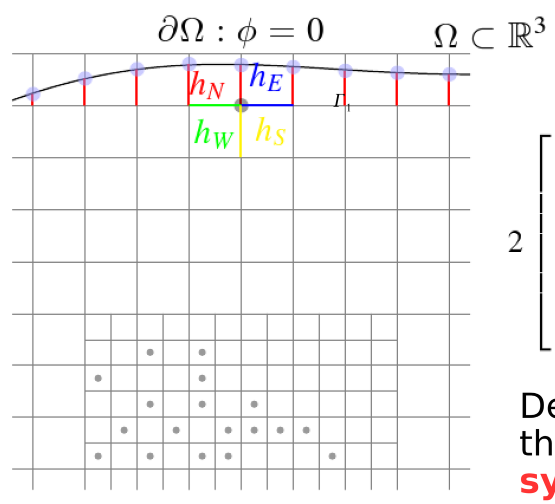
$$\nabla^2 \phi = -\frac{\rho}{\epsilon_0} \quad in\ \Omega \subset \mathbb{R}^3,$$

$$\phi = 0 \quad on\ \Gamma_1,$$

$$\frac{\phi}{\vec{n}} + \frac{1}{d}\phi = 0 \quad on\ \Gamma_2$$

Γ_1 is the surface of an elliptic or arbitrary beam-pipe
Γ_2 base areas at beginning and end of domain in *z*
d is the distance of the bunches centroid to the boundary

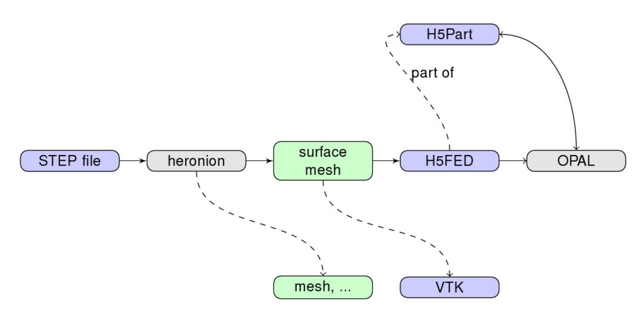
Shortley-Weller Approximation[5]



$$2\begin{bmatrix} & \frac{b}{h_N(h_N+h_S)} & \\ \frac{a}{h_W(h_W+h_E)} & -\frac{a}{h_W h_E} - \frac{b}{h_S h_N} & \frac{a}{h_E(h_W+h_E)} \\ & \frac{b}{h_S(h_N+h_S)} & \end{bmatrix}_h$$

Depending on boundary approximation the resulting discretization matrix is **non-symmetric**

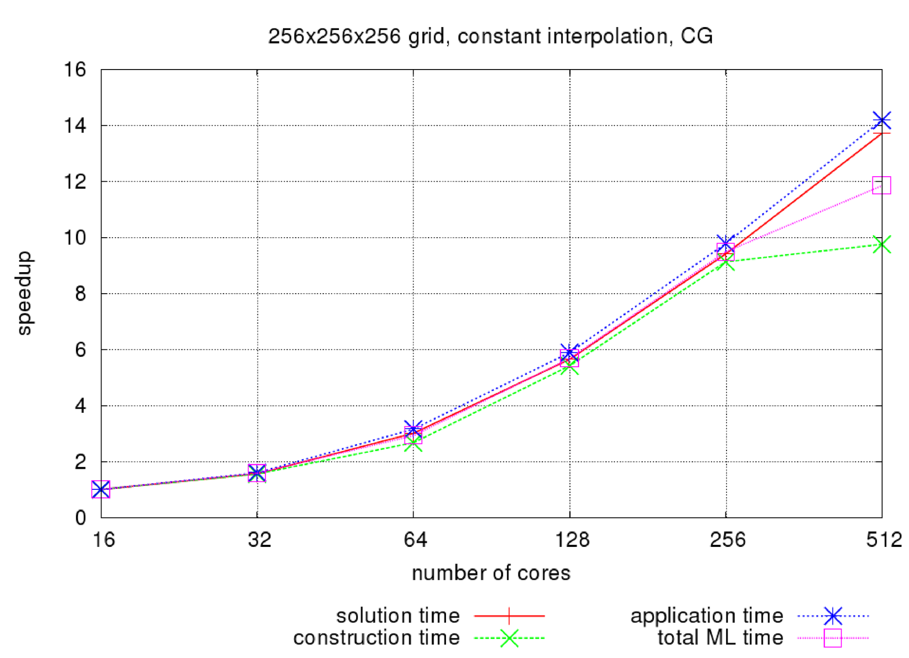
Workflow



RESULTS

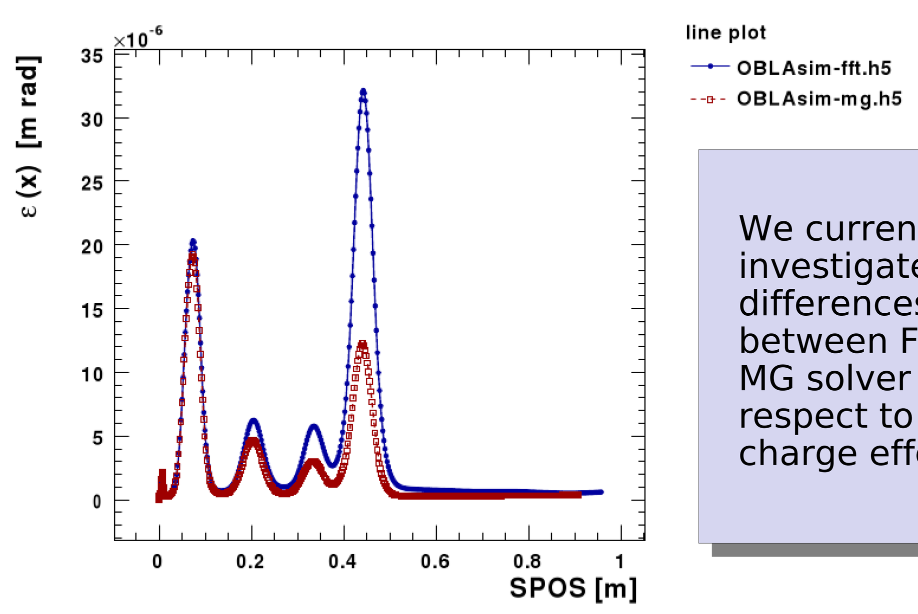
- Solver integrated into **OPAL**[6]
- Parallel in all directions
- Roughly 50% efficiency
- Validated against analytical solution
- New boundary condition result in reduced space-charge effects
- FFT solver 1.8x faster at the moment (MG on an elliptic bounded domain)

Speedup



- Small coarse level problem needs a lot communication
- Iterative coarse level solver: problem is dense, limit number of iterations

MG vs. FFT solver



We currently investigate differences between FFT and MG solver with respect to space-charge effects

References

[1] IPPL (Independent Parallel Particle Layer), Paul Scherrer Institute, http://amas.web.psi.ch/tools/IPPL
[2] Trilinos, Sandia National Laboratory, http://trilinos.sandia.gov
[3] Benedikt Oswald and Achim Gsell, H5fed, http://h5part.psi.ch
[4] T. Moeller and B. Trumbore, Fast, minimum storage ray/triangle intersection. Journal of Graphics Tools, 2(1):21–28, 1997
[5] G.H. Shortley and R. Weller, The numerical solution of Laplace's equation, Journal of Applied Physics, 9:334–344, 1938
[6] OPAL (Object Oriented Parallel Accelerator Library), PSI-PR-08-02, Paul Scherrer Institute, http://amas.web.psi.ch/tools/OPAL

Last update on 30.11.2008