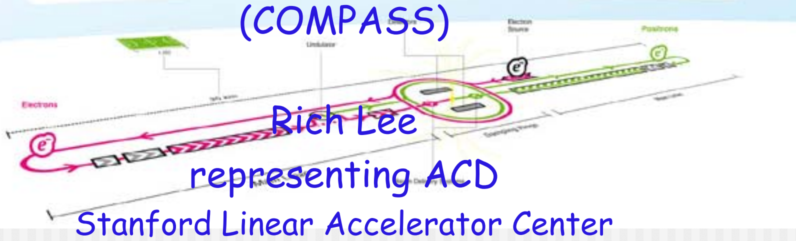
Finite-Element Electromagnetic Simulations for Particle Accelerators at Petascale

SciDAC Project: Community Petascale Project for Accelerator Science and Simulation (COMPASS)



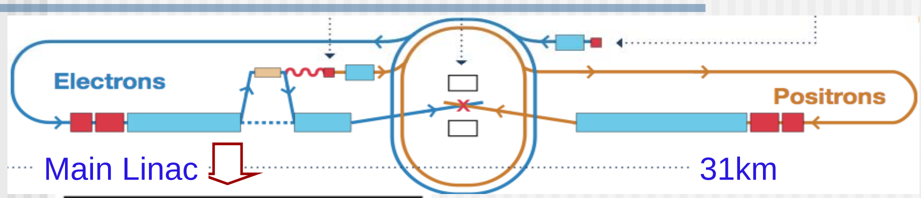
Rich Lee
representing ACD
Stanford Linear Accelerator Center

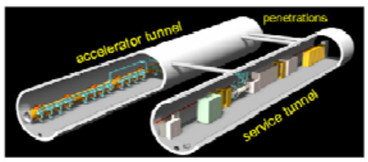
CScADs Workshop, July 2007

Overview

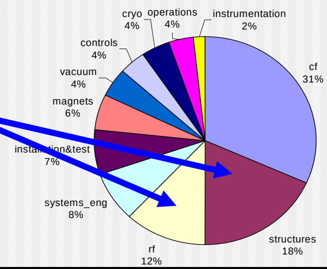
- International Linear Collider and Petascale Simulation Needs
- SLAC Electromagnetic Efforts
  - Team
  - Codes
- Computational Methods
- Software Design
- Performance and Scalability
- Current AM/CS Collaborations
- Additional Challenges to reach Petascale

ILC Superconducting Main Linac

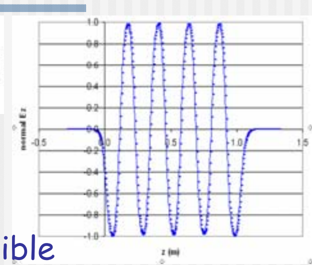




SRF Main Linac constitutes the heart of the accelerator at 30% of its total cost & consists of 20,000 SRF cavities to accelerate the beams to 0.5 TeV energy



Accelerator Cavity Design and Simulations



Accelerating Mode:

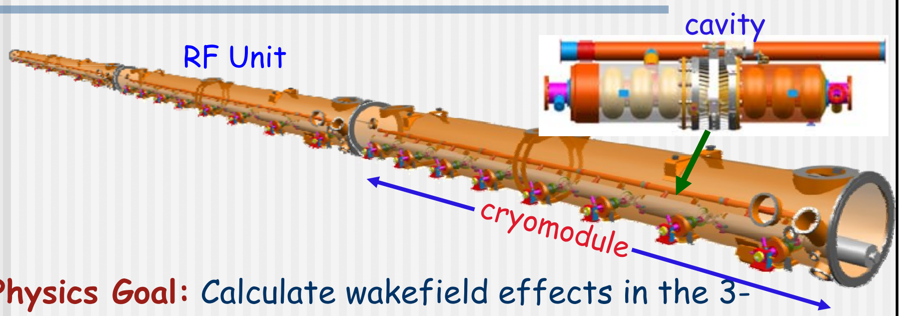
- Pi mode at frequency – 1.3 GHz
- Field flatness
- Accelerating gradient – highest possible

Wakefields (Higher-order-modes or HOMs):

Parasitic electromagnetic fields generated by the leading bunch that affect the tail of the bunch or following bunches degrade beam transport (emittance) down the linacs.

- Measurement is difficult so one relies heavily on simulation
- Computations carried out to determine damping effects

Petascale Simulation Need: Modeling RF Unit of ILC Main Linac



Physics Goal: Calculate wakefield effects in the 3-cryomodule RF unit (26 cavities) with realistic 3D dimensions and misalignments

- Trapped mode and damping
- Cavity imperfection effects on HOM damping
- Wakefield effect on beam dynamics
- Effectiveness of beam line absorber

SLAC Electromagnetics Team

Advanced Computations Department @ SLAC

Accelerator Modeling	Computational Mathematics	Computing Technologies
K. Ko, C. Ng, A. Candel, A. Kabel, Z. Li, L. Xiao	*L. Lee, V. Akcelik, L. Ge, E. Prudencio, S. Chen (Stanford)*	*G. Schussman, R. Uplenchwar, S. Czech*

SciDAC CETs / Institutes (TOPS, ITAPS, CSCAPES, IUSV, PERI)

LBNL	LLNL	SNL	ORNL
E. Ng, X. Li, C, Yang P. Husbands, A. Pinar, D. Bailey, D. Gunter	*L. Diachin, D. Quinlan, R. Vuduc*	*P. Knupp, J. Kraftcheck K. Devine. E. Boman*	*R. Barrett*

ANL/UW	UCDavis	UT	RPI	Columbia
T. Tautges	*K. Ma, H. Yu, Z. Bai, B. Liao*	*O. Ghattas*	*M. Shepard, X. Luo, A. Brewer*	*D. Keyes*

SLAC's Parallel Electromagnetic Codes

*Solve Maxwell's equations in time & frequency domains using **3D unstructured grid** and **parallel computing***

Finite-element Basis up to 6th order

- T3P → Time Domain Simulation with Excitations
- Omega3P → Frequency Domain Mode Analysis
- S3P → Scattering Matrix Evaluation
- Pic3P → Self-Consistent Particle-in-Cell
- TEM3P → Integrated Thermal Mechanical EM

Track3P – Particle Tracking with surface physics

V3D – Visualization of Meshes, Particles & Fields

RF Cavity Eigenvalue Problem

Find frequency and field vector of normal modes:

Γ_E ***Closed Cavity*** Γ_M

"Maxwell's Eqns in Frequency Domain"

$$\nabla \times \left(\frac{1}{\mu}\nabla \times \vec{\mathbf{E}}\right) - k^2 \epsilon \vec{\mathbf{E}} = 0 \ on \ \Omega$$

$$\vec{\mathbf{n}} \times \vec{\mathbf{E}} = 0 \ on \ \Gamma_E$$

$$\vec{\mathbf{n}} \times \frac{1}{\mu}\nabla \times \vec{\mathbf{E}} = 0 \ on \ \Gamma_M$$

Nedelec-type Element $\mathbf{E} = \sum_i x_i \mathbf{N}_i$

$$\mathbf{Kx} = k^2 \mathbf{Mx}$$

$$\mathrm{K}_{ij} = \int_\Omega (\nabla \times \mathrm{N}_i) \cdot \frac{1}{\mu} (\nabla \times \mathrm{N}_j)\, d\Omega$$

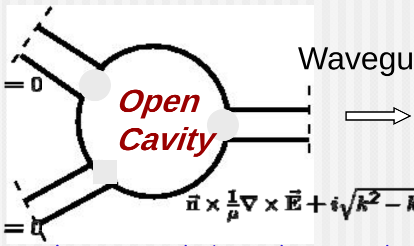
$$\mathbf{M}_{ij} = \int_\Omega \mathbf{N}_i \cdot \epsilon \mathbf{N}_j \, d\Omega$$

Cavity with Waveguide Coupling

- One waveguide mode per port only

Waveguide BC

$\vec{n} \times \frac{1}{\mu}\nabla \times \vec{E} + i\sqrt{k^2 - k_{c_3}^2}\,\vec{n} \times \vec{n} \times \vec{E} = 0$



Waveguide BC

Waveguide BC

$\vec{n} \times \frac{1}{\mu}\nabla \times \vec{E} + i\sqrt{k^2 - k_{c_2}^2}\,\vec{n} \times \vec{n} \times \vec{E} = 0$

$\vec{n} \times \frac{1}{\mu}\nabla \times \vec{E} + i\sqrt{k^2 - k_{c_1}^2}\,\vec{n} \times \vec{n} \times \vec{E} = 0$

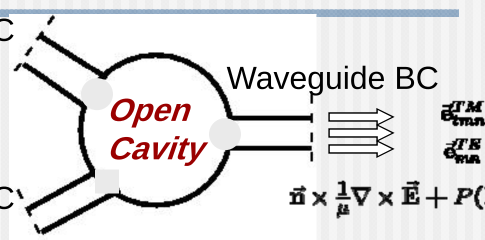
- Vector wave equation with waveguide boundary conditions can be modeled by a **non-linear eigenvalue problem**

$$\mathbf{K}x + i\sum_j \sqrt{k^2 - k_{cj}^2}\,\mathbf{W}_j x = k^2\mathbf{M}x$$

With $(\mathbf{W}_j)_{ik} = \int_\Gamma (n \times \mathbf{N}_i)\cdot(n \times \mathbf{N}_k)\,d\Gamma$

Cavity with Waveguide Coupling for Multiple Waveguide Modes

Waveguide BC



Waveguide BC

Waveguide BC

$\vec{n} \times \frac{1}{\mu}\nabla \times \vec{E} + P(\vec{E}) = 0$ where

$$P(\vec{E}) = \sum_m^\infty \sum_n^\infty \frac{k^2}{i\sqrt{k^2 - k_{mn}^2}}\vec{e}^{TM}_{tmn}\int_\Gamma \vec{e}^{TM}_{tmn}\cdot\vec{E}\,d\Gamma - \sum_m^\infty \sum_n^\infty i\sqrt{k^2 - k_{mn}^2}\,\vec{e}^{TE}_{mn}\int_\Gamma \vec{e}^{TE}_{mn}\cdot\vec{E}\,d\Gamma$$

- Vector wave equation with waveguide boundary conditions can be modeled by a **non-linear eigenvalue problem**

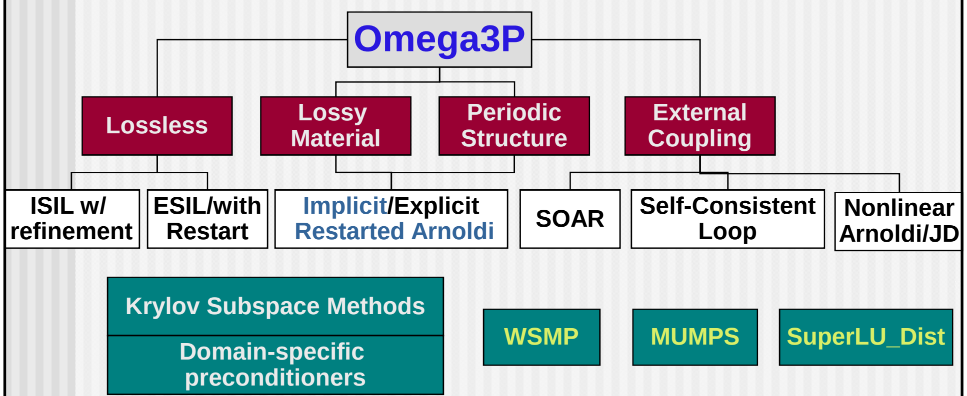
$$\mathbf{K}x + i\sum_{m,n}\sqrt{k^2 - k_{mn}^2}\,\mathbf{W}^{TE}_{mn}x + i\sum_{m,n}\frac{k^2}{\sqrt{k^2 - k_{mn}^2}}\mathbf{W}^{TM}_{mn}x = k^2\mathbf{M}x$$

where

$$(\mathbf{W}^{TE}_{mn})_{ij} = \int_\Gamma \vec{e}^{TE}_{mn}\cdot\mathbf{N}_i\,d\Gamma \int_\Gamma \vec{e}^{TE}_{mn}\cdot\mathbf{N}_j\,d\Gamma$$

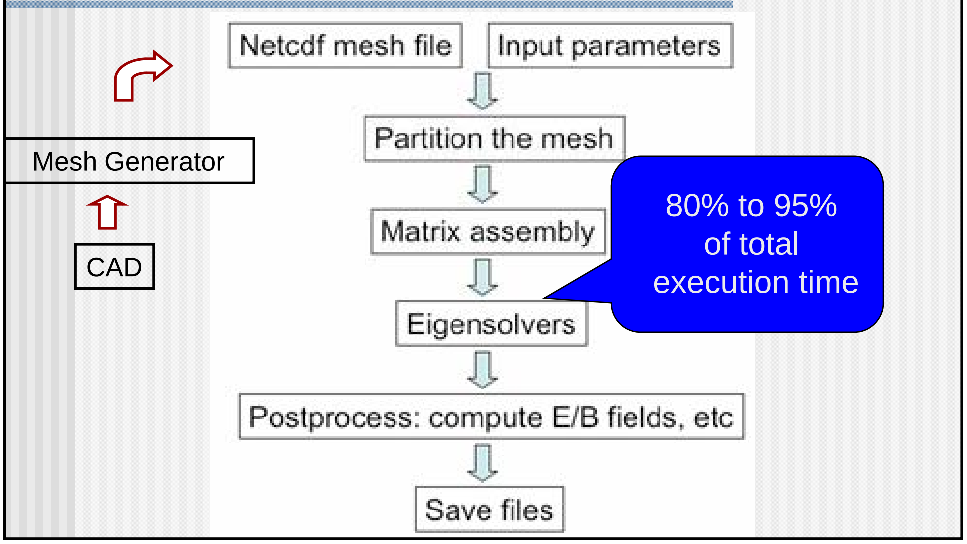
$$(\mathbf{W}^{TM}_{mn})_{ij} = \int_\Gamma \vec{e}^{TM}_{tmn}\cdot\mathbf{N}_i\,d\Gamma \int_\Gamma \vec{e}^{TM}_{tmn}\cdot\mathbf{N}_j\,d\Gamma$$

Problems and Solver Options

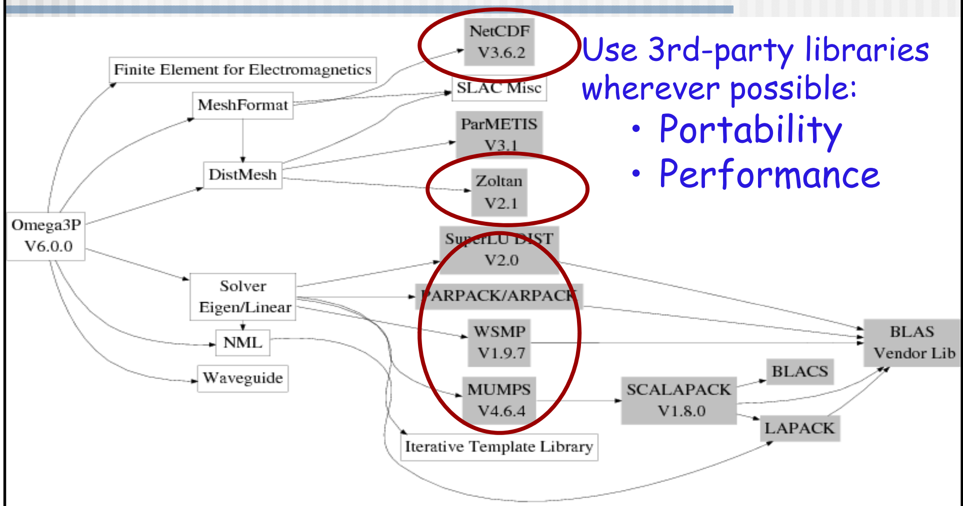


Different solver options have different performance dynamics

Omega3P Execution Flow

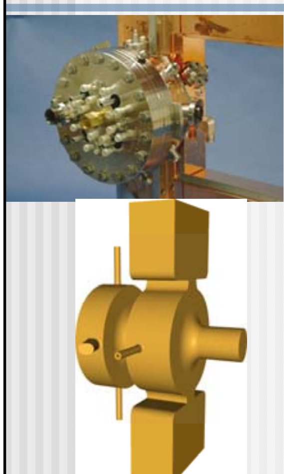


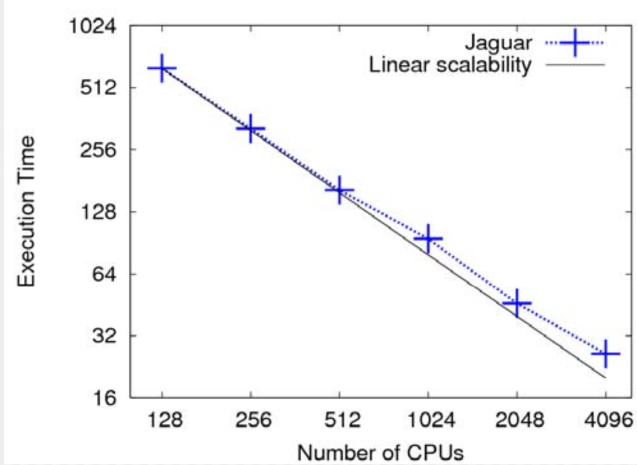
Components-Based Software Design



Use 3rd-party libraries wherever possible:
- Portability
- Performance

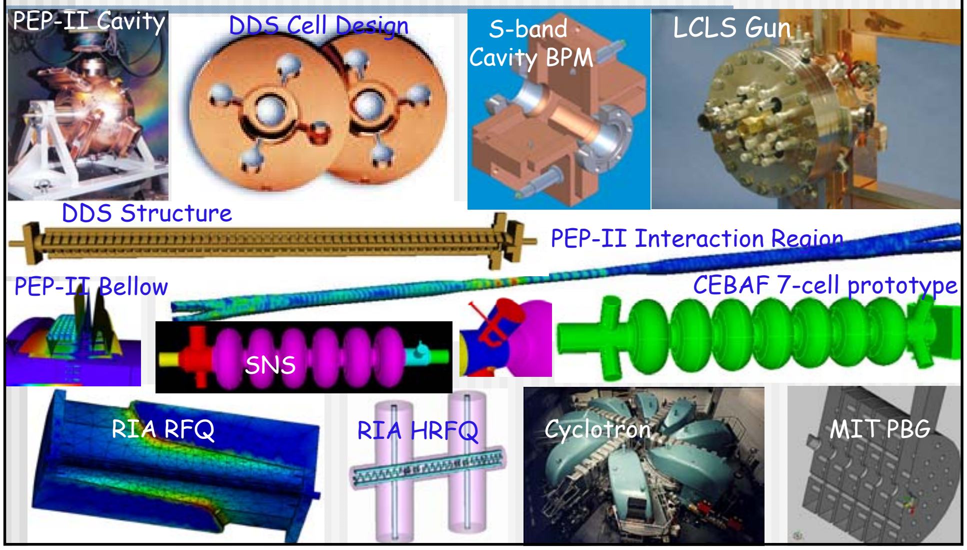
Omega3P Scalability on Jaguar/XT (LCLS RF gun)



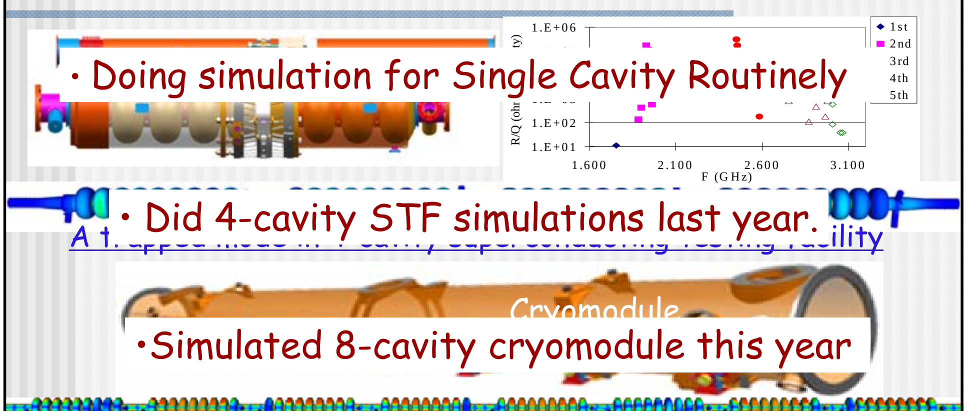


- 1.5M tetrahedral elements   NDOFs = 9.6M   NNZ = 506M

Omega3P Accomplishments (Accelerator Applications)



Model and Design ILC Cavities, Superstructures, Cryomodules

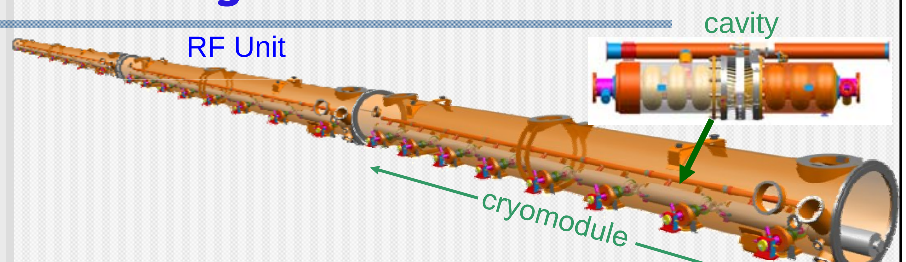


- Doing simulation for Single Cavity Routinely
- Did 4-cavity STF simulations last year.
- Simulated 8-cavity cryomodule this year

A dipole mode in 8-cavity cryomodule at 3rd band

First ever calculation of a 8 cavity cryomodule
- *~ 20 M DOFs*
- *~ 1 hour per mode on 1500 CPUs for the cryomodule*

Modeling RF Unit of ILC Main Linac



Physics Goal: *Calculate wakefield effects in the 3-cryomodule RF unit with realistic 3D dimensions and misalignments*

To model a 3-module RF unit would require

- *>200 M DOFs*
- *Advances in algorithm and solvers*
- *Petascale computing resources*

Finite-Element Time-Domain Code: T3P

Time-domain second-order vector wave equation:

$$\nabla \times \left(\frac{1}{\mu}\nabla \times \vec{\mathrm{E}}\right) + \sigma\frac{\partial \vec{\mathrm{E}}}{\partial t} + \epsilon\frac{\partial^2 \vec{\mathrm{E}}}{\partial t^2} = -\frac{\partial \vec{\mathrm{J}}}{\partial t} \ \textit{on}\ \Omega$$

$$\vec{\mathbf{n}} \times \vec{\mathbf{E}} = 0 \ \textit{on}\ \Gamma_E$$

$$\vec{\mathbf{n}} \times \frac{1}{\mu}\nabla \times \vec{\mathbf{E}} = 0 \ \textit{on}\ \Gamma_M$$

Nedelec-type Element $\vec{\mathbf{E}} = \sum_i x_i(t)\mathbf{N}_i(\vec{\mathbf{r}})$

$$\mathrm{M}\frac{1}{c^2}\frac{\partial^2 \mathrm{x}}{\partial t^2} + (\mathrm{R} + \mathrm{Q})\frac{1}{c}\frac{\partial \mathrm{x}}{\partial t} + \mathrm{Kx} = \mathrm{f}$$

$$\mathbf{K}_{ij} = \int_\Omega (\nabla \times \mathbf{N}_i)\cdot\frac{1}{\mu}(\nabla \times \mathbf{N}_j)\,d\Omega \qquad \mathbf{R}_{ij} = \int_\Omega \mathbf{N}_i \cdot \sigma\mathbf{N}_j\,d\Omega$$

$$\mathbf{M}_{ij} = \int_\Omega \mathbf{N}_i \cdot \epsilon\mathbf{N}_j\,d\Omega \qquad \mathbf{Q}_{ij} = \int_\Gamma (\mathbf{n} \times \mathbf{N}_i)\cdot\frac{1}{\mu}(\mathbf{n} \times \mathbf{N}_j)\,d\Gamma$$

$$\mathbf{f}_i = -\int_\Omega \mathbf{N}_i \cdot \frac{\partial \mathbf{J}}{\partial t}\,d\Omega$$

Newmark-β Scheme for T3P

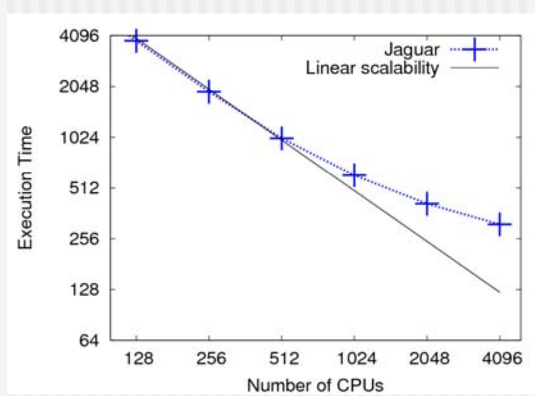
$$\left(\mathbf{M} + \frac{c\Delta t}{2}(\mathbf{R}+\mathbf{Q}) + \beta(c\Delta t)^2\mathbf{K}\right) x^{n+1} = b$$

$$\begin{aligned} b = & \,(2\mathbf{M} - (1-2\beta)(c\Delta t)^2\mathbf{K})x^n \\ & - \left(\mathbf{M} - \tfrac{1}{2}c\Delta t(\mathbf{R}+\mathbf{Q}) + \beta(c\Delta t)^2\mathbf{K}\right) x^{n-1} \\ & - (c\Delta t)^2(\beta f_{n+1} + (1-2\beta)f^n + \beta f_{n-1}) \end{aligned}$$

- Unconditionally stable when $\beta > 0.25$
- Matrix in the linear system is SPD
- Performance/Scalability largely depends on solving linear system and computing b

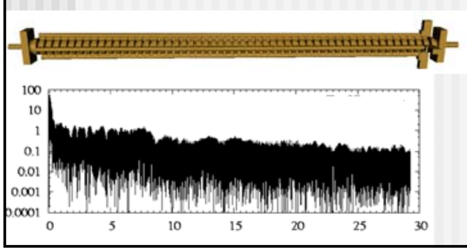
T3P Scalability on Jaguar/XT (ILC TDR Cavity)

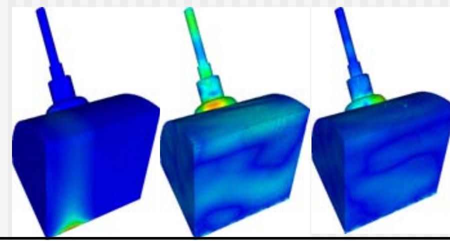
- Beam passes through cavity
- 1.81 million tetrahedral mesh
- NDOFs=11.2 M
- Δt=4ps
- Run up to 6ns



T3P Accomplishments

- First direct verification of NLC DDS Wakefield
- ILC TDR Cavity Wakefield (movie)
- ILC Superconducting Testing Facility Wakefield
- ILC Damping Ring Beam Position Monitor (BPM)





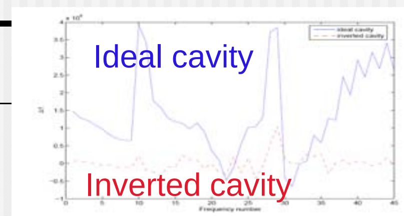
Current SciDAC CS/AM Collaborations

- Shape Determination & Optimization (TOPS/UT Austin, LBNL) –
  – *Obtain cavity deformations from measured mode data through solving a weighted least square minimization problem*
- Parallel Complex Nonlinear Eigensolver/Linear Solver (TOPS/LBNL)
  – *Develop scalable algorithms for solving LARGE, complex, nonlinear eigenvalue problems to find mode damping in the rf unit complete with input/HOM couplers and external beampipes*
- Parallel Adaptive Mesh Refinement and Meshing (ITAPS/RPI, ANL)
  – *Optimize computing resources and increase solution accuracy through adaptive mesh refinement using local error indicator based on gradient of electromagnetic energy in curved domain*
- Parallel Load Balancing (ITAPS,CSCAPES)
  – *Balance the load for field-particle coexisting simulations such as PIC*
- Parallel and Interactive Visualization (ISUV/UC Davis) –
  – *Visualize complex electromagnetic fields and particles with large complex geometries and large aspect ratio*

Solver Collaborations between SLAC and TOPS

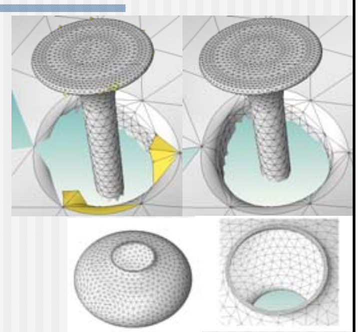
- Effective domain-specific multilevel preconditioner
- Solver for KKT linear system
- "Memory-usage" scalable solvers
- AMLS for linear and nonlinear eigensystems
- Shape Determination with Noisy Data

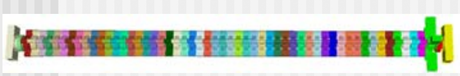
Method	Time (s)	Memory (GB)
MUMPS	293.8	155.3
MUMPS with single precision	450.1	82.3

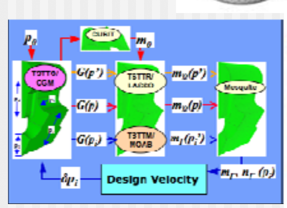


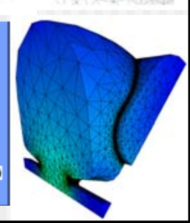
Working with ITAPS

- Tool for correcting inverted 10-point Tetrahedron
- Anisotropic mesh for integrated multiphysics analysis
- Parallel AMR/Meshing
- Shape derivative
- Moving AMR for time-domain simulations

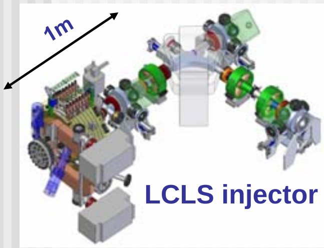


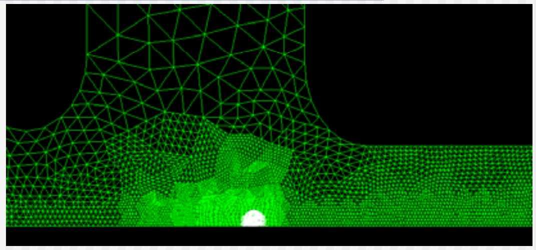


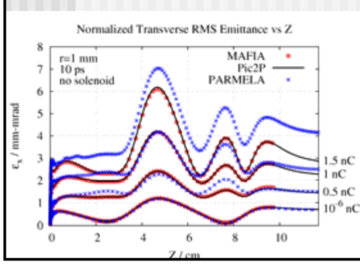




Load Balancing for ITAPS and CSCAPES



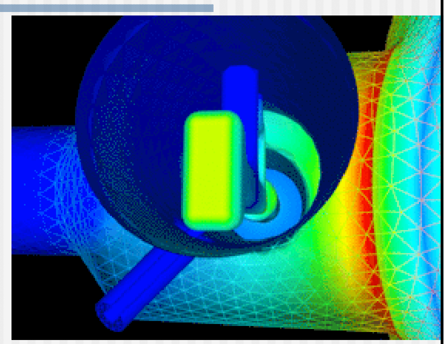




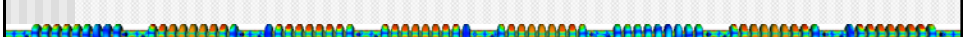
- 1st successful self-consistent, charge-conserving PIC code with conformal Whitney elements on unstructured FE grid
- Need to Balance different loads: field computation and particle pushing

Working with IUSV

- Vector Fields, Particles, Complex Geometries
- Large Data
- Large aspect ratio



Need Parallel, Interactive and Remote Visualization

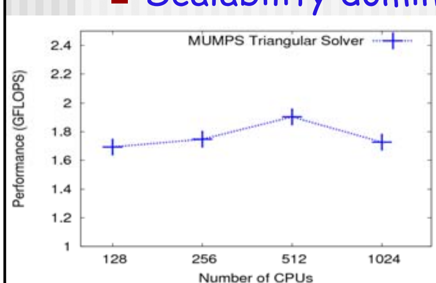


Additional Challenges to Reach Petascale

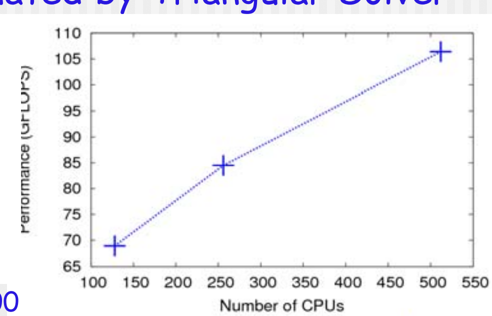
- Faster sparse matrix-vector kernel on specific petascale computing platforms
- More scalable triangular solver in Sparse Direct Solvers
- More memory-usage scalable Sparse Direct Solver
- Memory-usage debugging tool
- Translation tool? (MPI --> MPI+OpenMP)
- Common performance analysis tool
- Parallel IO

Scalability Using Sparse Direct Solver MUMPS

- Sparse Direct Solver is effective for highly indefinite matrices
- Scalability dominated by Triangular Solver



N=2,019,968, nnz=32,024,600
No. of entries in L =1 billion

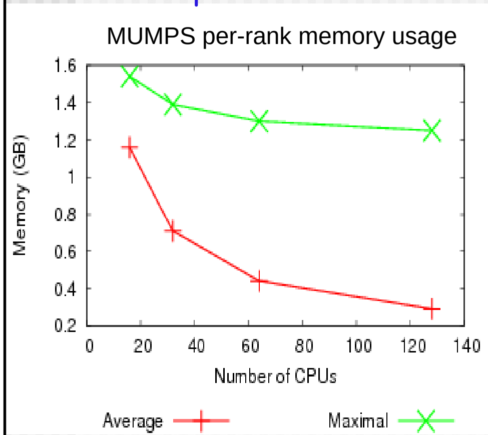


N=2M, PSPASES Triangular Solver

- Need a more scalable Triangular Solver

More "*Memory-usage*" Scalable Sparse Direct Solvers

N=1.11M, nnz=46.1M
Complex matrix



- Maximal per-rank MU is 4-5 times than the average MU
- Once it cannot fit into Nprocs, it most likely will not fit into 2*Nprocs
- More "memory-usage" scalable solver needed

Memory-usage Debugging Tool

- 0.5 to 4GB per-node, no virtual memory
- Exceed the amount of physical memory -> failed job
- Need to squeeze every single byte
- Example: C++ allocators are not friendly

```
#include <iostream>
#include <map>
#include <vector>
int main() {
  using namespace std;
  cout << "creating map now" << endl;
  int size=10000000;  //10 million
  {
    map<int,int> mm;
    for (int i=0; i<size; i++)  mm[i]=i;
    mm.clear();
  }
  cout << "map should no longer use memory"
       << endl;
  while(1);
  return 0;
}
```

IO Handling through NetCDF (Write)

Method 1:

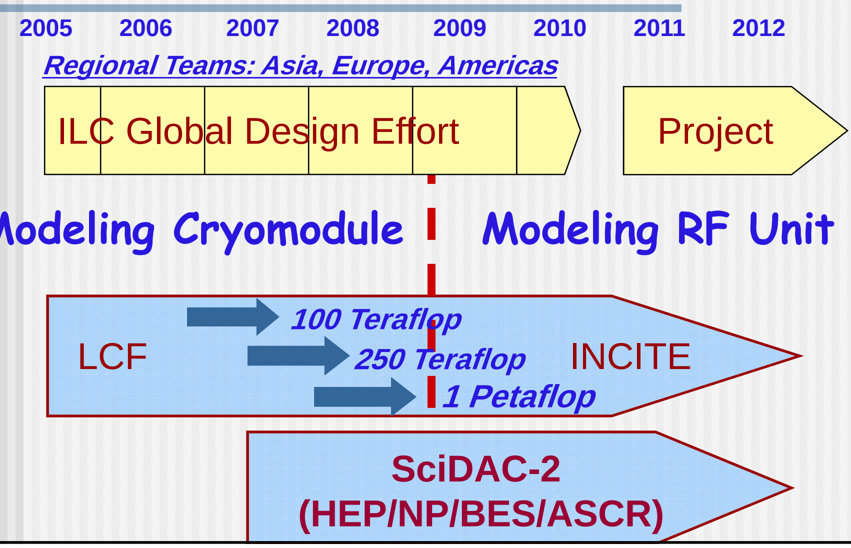
- Rank 0 create netcdf file and close
- Each rank open the file and write his own portion sequentially, close
- Highly rely on FS
- Trouble on NFS-type with Nprocs > 2000

Method 2:

- Rank 0 collect data and handle all the writing
- Robust but scalability bottleneck for Nprocs >= 2000

Will the use of parallel-netcdf solve the problem?

ILC, Petascale Computing & SciDAC



Thanks

CScADs Workshop, July 2007