Sparse Matrix Computation in Large-Scale Scientific Applications

Esmond G. Ng
Lawrence Berkeley National Laboratory

June 16, 2010

Scientific Computing @ U.S. Department of Energy

- The U.S. Department of Energy (DOE) is one of several federal agencies that fund basic and applied scientific research.
- DOE supports a large variety of research and development in modeling, simulation, and computation, particularly in National Nuclear Security Administration (NNSA) and the Office of Science.
- DOE also funds large-scale computing facilities that provide high performance computing resources.
  - Examples:
    - NERSC – National Energy Research Scientific Computing Center (Lawrence Berkeley National Laboratory)
    - LCF – Leadership Computing Facility (Oak Ridge National Laboratory)
    - ALCF – Argonne Leadership Computing Facility (Argonne National Laboratory)
    - ESnet – Energy Sciences Network (Lawrence Berkeley National Laboratory)
    - Terascale Simulation Facility (Lawrence Livermore National Laboratory)
    - …

DOE Computing Facilities *(October 2009 statistics)*

IBM Cluster (Roadrunner) @ LANL
NNSA ; #1 on TOP500
6,120 AMD dual-core Opterons
12,240 IBM Cell processors
Theoretical peak (Cell) = 1.33 PFlop/sec
Total physical memory = 98 TB

Cray XT-4/XT-5 (Jaguar) @ ORNL
Office of Science ; #2 on TOP500
Two partitions (XT-4 and XT-5)
XT-5 partition
No. of nodes = 18,688
Processor cores per node = 8
No. of compute processor cores = 149,504
Theoretical peak = 1.38 PFlop/sec
Physical memory per compute node = 16 GB

Blue Gene/P (Intrepid) @ ANL
Office of Science ; #7 on TOP500
No. of nodes = 40,960
Processor cores per node = 4
No. of compute processor cores = 163,840
Theoretical peak = 557 TFlop/sec
Total physical memory = 80 TB

Cray XT-4 (Franklin) @ Berkeley Lab
Office of Science ; #11 on TOP500
No. of nodes = 9,572
Processor cores per node = 4
No. of compute processor cores = 38,288
Theoretical peak = 352 TFlop/sec
Physical memory per compute node = 8 GB

The next machine to be installed in 2010 will be a Cray XT-5, with 24 processor cores per node and a peak performance of 1.17 PFlop/sec

DOE Large-Scale Scientific Applications

- Strong emphasis on high-end computational sciences at DOE – many are large, multi-institutional projects.
  - Accelerators, astrophysics, nuclear physics
  - Chemistry
  - Fusion
  - Bioremediation, groundwater flows
  - Climate
  - …
- DOE also funds large projects in applied mathematics and computer science.
  - The main goal is the development of new high-performance scalable algorithms/tools for core components of scientific simulation, and the distribution of those algorithms/tools through portable high-performance libraries.
  - Also help scientific applications to effectively utilize the massively parallel computers.

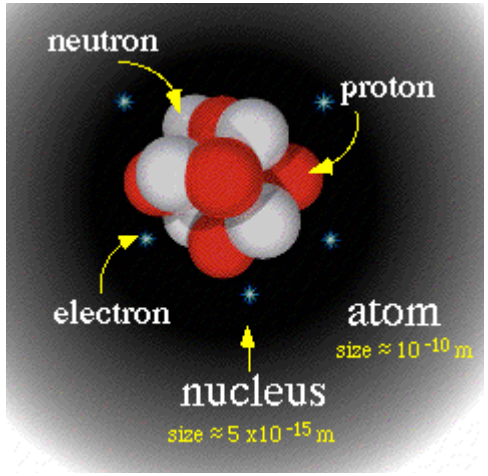
DOE Scientific Applications

- Most of the scientific applications are PDE-based.
  - The innermost kernels are often linear algebra problems.
  - The majority of the linear algebra problems are
    - Large sparse linear systems
    - Large sparse eigenvalue calculations

- Consider 2 examples …
  - Nuclear structure calculations
  - Accelerator modeling

Nuclear Structure Calculations

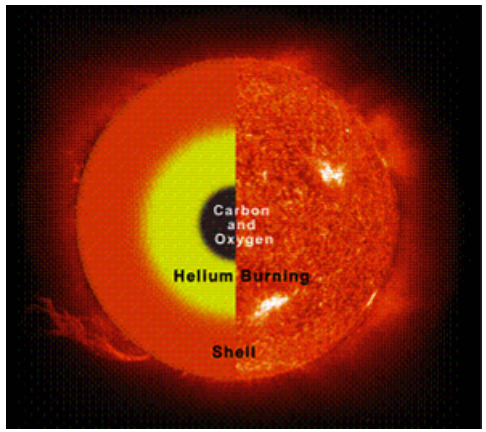
- Determine the microscopic structure of nuclei, and the strong interactions among protons and neutrons.
  - Original of the ^{12}C formation in stars.
  - Foundation for nuclear reaction theory.



- The quantum many-body problem is described by the nuclear Schrödinger equation

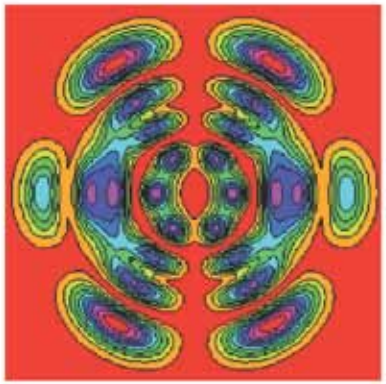
$$\mathbf{H}\Psi(r_1, r_2, \ldots, r_k) = \lambda\Psi(r_1, r_2, \ldots, r_k)$$

  - H – nuclear Hamiltonian describes kinetic energy, as well as 2-body and 3-body potentials;
  - Ψ – nuclear wavefunction; $|\Psi(r_1,r_2,\ldots,r_k)|^2$ – probability density of finding nucleons 1, 2, …, k at r_1, r_2, …, r_k;
  - λ – quantized energy level. Often interested in the ground state (λ_1) and a few (10-100) low excited states

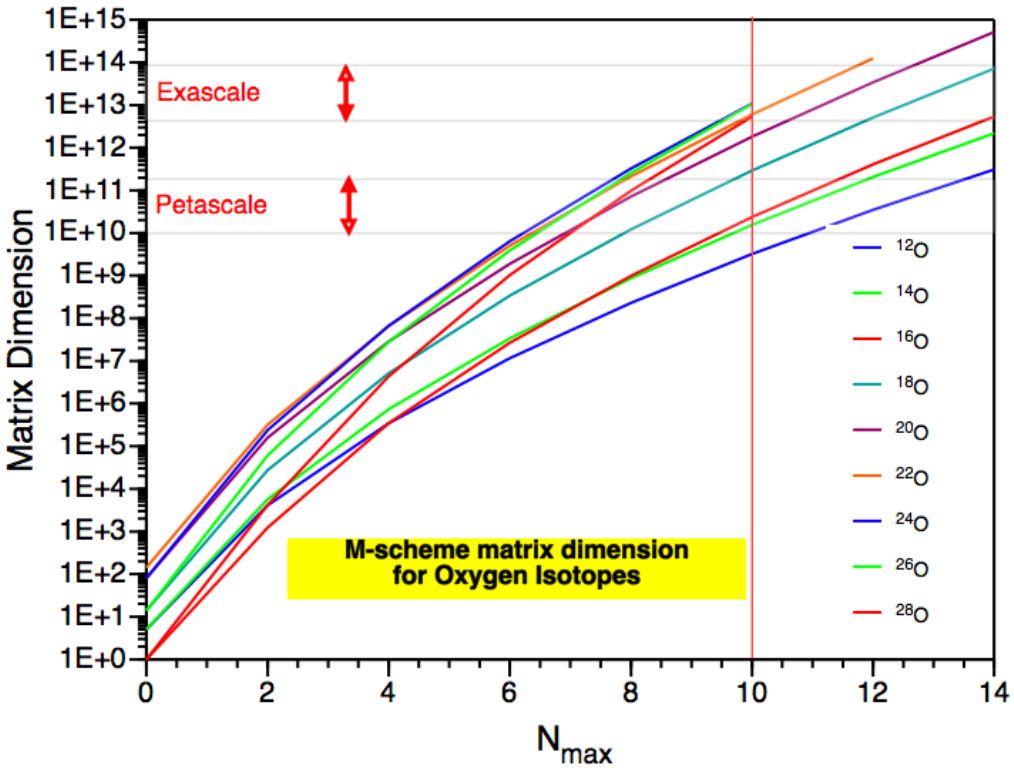


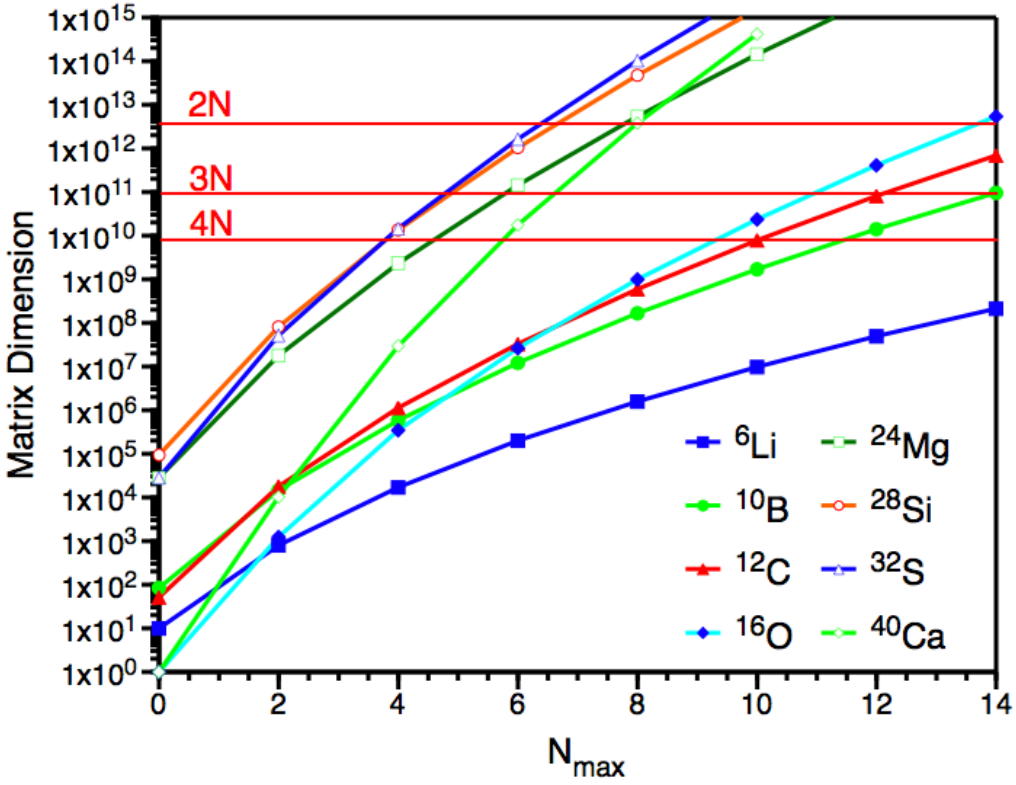
Nuclear Structure Calculations

- Solving the many-body problem directly is not feasible.
- Using "ab initio" no-core shell model and full configuration interaction methods:
  - Calderon, Ng, Sternberg, Yang + Sosonkina, Maris, Vary
  - Expand the wavefunctions using some chosen basis.
    - Typically a harmonic oscillator basis.
  - The problem reduces to a symmetric eigenvalue problem $Hx = \lambda x$.
  - Dimension of H, which depends in part on the size of the basis space, can be very large but quite sparse.
  - Sparsity depends on NN or NN + NNN interactions.
  - For example, ^{16}O, $N_{max} = 8$, 2-body & 3-body potentials:
    - Dimension $\approx 10^8$
    - # of nonzero entries $\approx 8 \times 10^{11}$

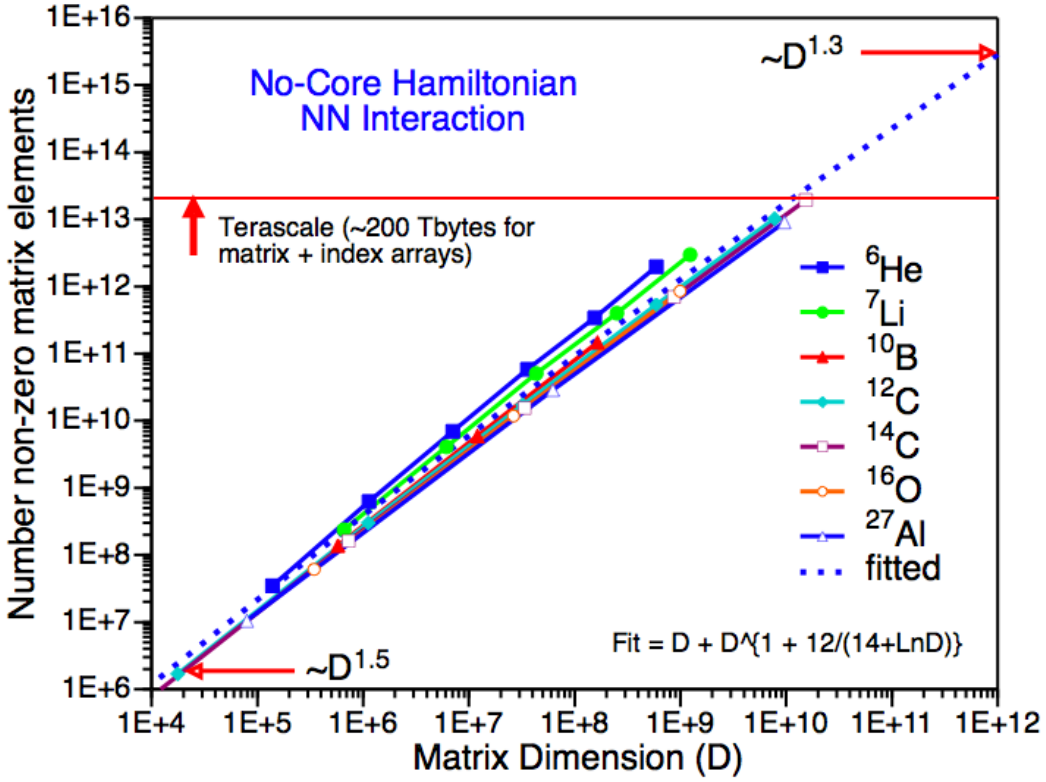


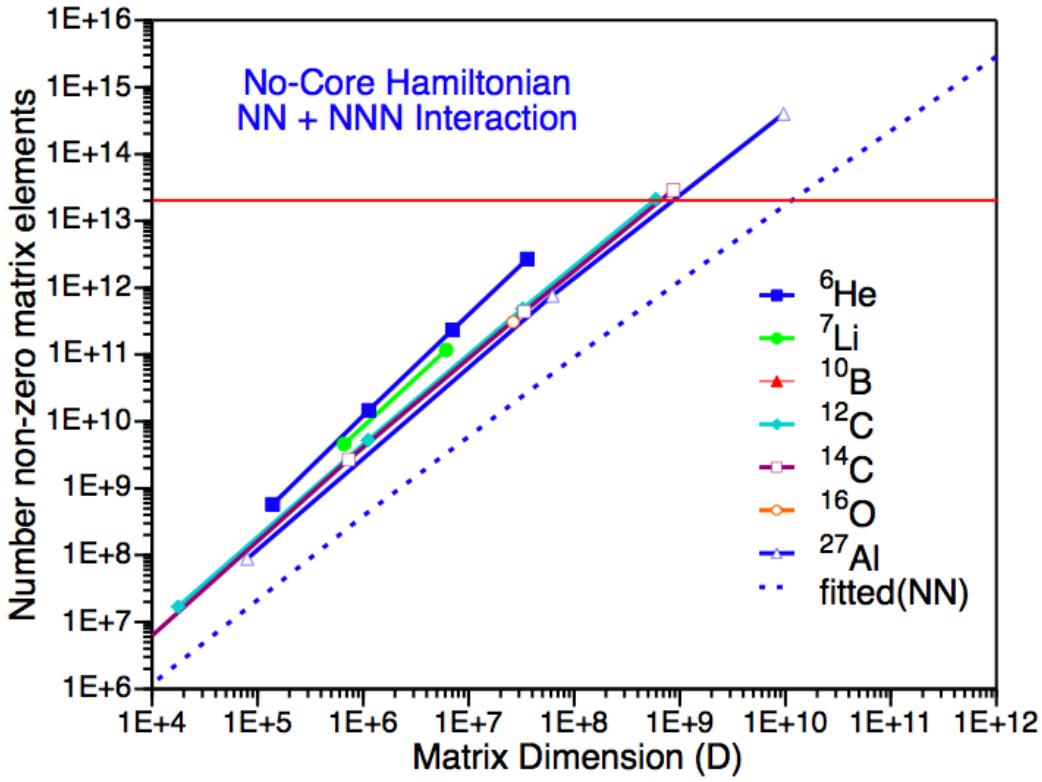
Nuclear Structure Calculations





Nuclear Structure Calculations





Eigenvalue Calculations in Nuclear Structure

- Using the Lanczos algorithm …
  - Fully parallel, running on significant portion of Jaguar
  - Balanced workload

- Bottlenecks
  - Construction of the Hamiltonian matrix
    - Matrix depends on choice N_{max}.
    - Sparsity structure is determined on the fly.
  - Performance of matrix-vector multiplications
    - Choice of indexing and data structure.

Construction of the Hamiltonian Matrix

- Rows/columns indexed by many-body states

$$s = \overbrace{(s_1, s_2, \cdots, s_N)}^{\text{many-body state}} \quad : \quad s_i < s_{i+1}$$

- s_i's are single-particle states.
- Physics excludes most of the $\begin{pmatrix} N_{spstates} \\ N_{nucleons} \end{pmatrix}$ many-body states.
- If *s* and *t* are many-body states that differ by more than two single-particle states (with 2-body potentials), the matrix element indexed by *s* and *t* is exactly zero.
- If not, we call *s* and *t* an interacting pair.

Characterization of Nonzero Entries

- Example - Consider 3 many-body states.

a = { 2 , 3 , 4 , 7 , 9 , 12 }

b = { 1 , 2 , 4 , 7 , 8 , 12 }

c = { 1 , 4 , 5 , 7 , 8 , 9 }

(a , c) is not an interacting pair

(a , b) is an interacting pair

(b , c) is an interacting pair

Characterization of Nonzero Entries

- Example - Consider 3 many-body states.

a = { 2 , 3 , 4 , 7 , 9 , 12 }
b = { 1 , 2 , 4 , 7 , 8 , 12 }
c = { 1 , 4 , 5 , 7 , 8 , 9 }

(a , c) is not an interacting pair
(a , b) is an interacting pair
(b , c) is an interacting pair

Characterization of Nonzero Entries

- Example - Consider 3 many-body states.

a = { 2 , 3 , 4 , 7 , 9 , 12 }

b = { 1 , 2 , 4 , 7 , 8 , 12 }

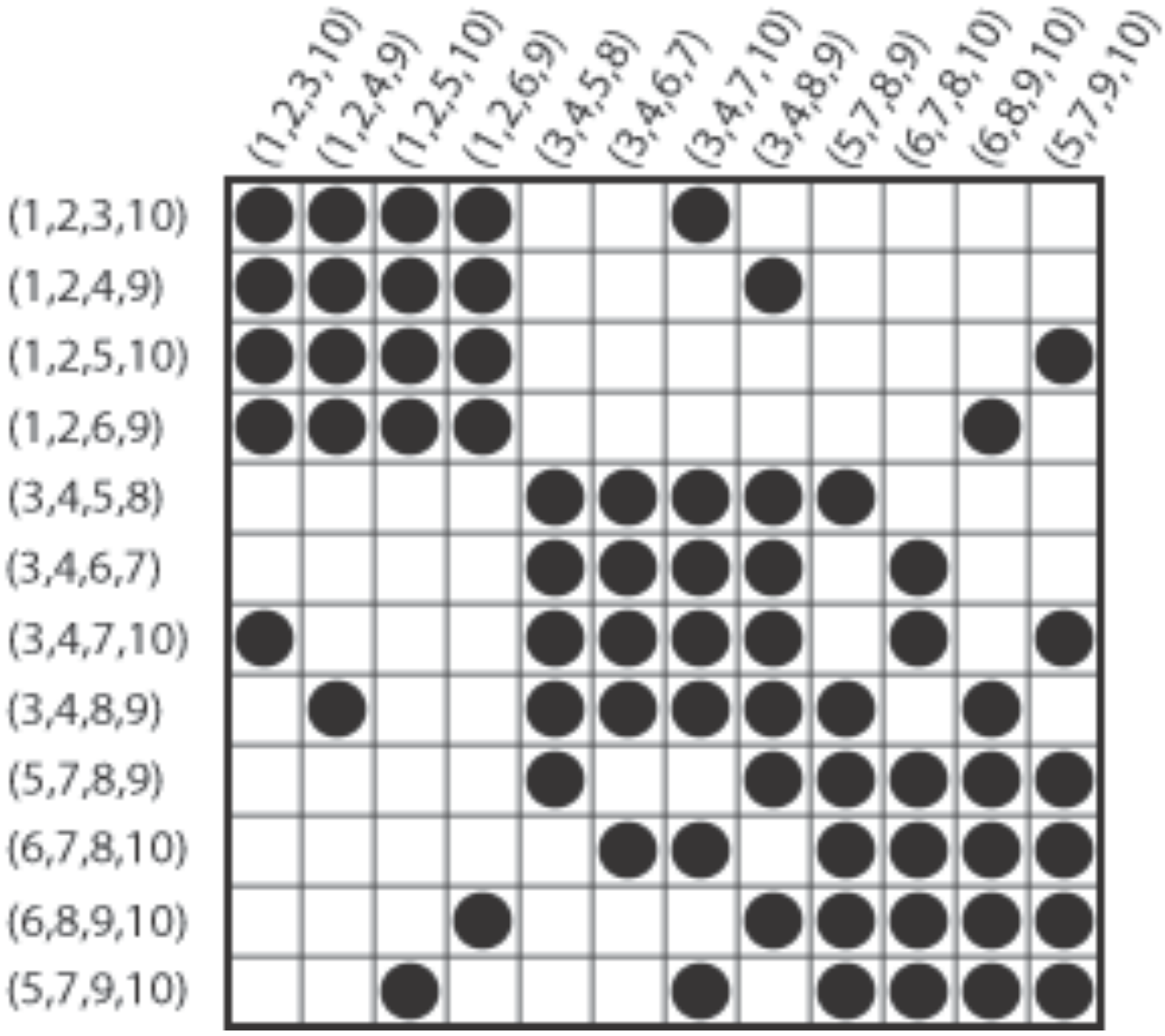
c = { 1 , 4 , 5 , 7 , 8 , 9 }

(a , c) is not an interacting pair

(a , b) is an interacting pair

(b , c) is an interacting pair

Tiny Example



The Need for Blocking

- Exhaustive pairwise comparison is prohibitively expensive.

- Use clustering to identify large zero blocks [Sternberg].
- Partition the single-particle states into bins, then cluster the many-body states based on how many single-particle states are in each bin.

The Need for Blocking

Example: Partition

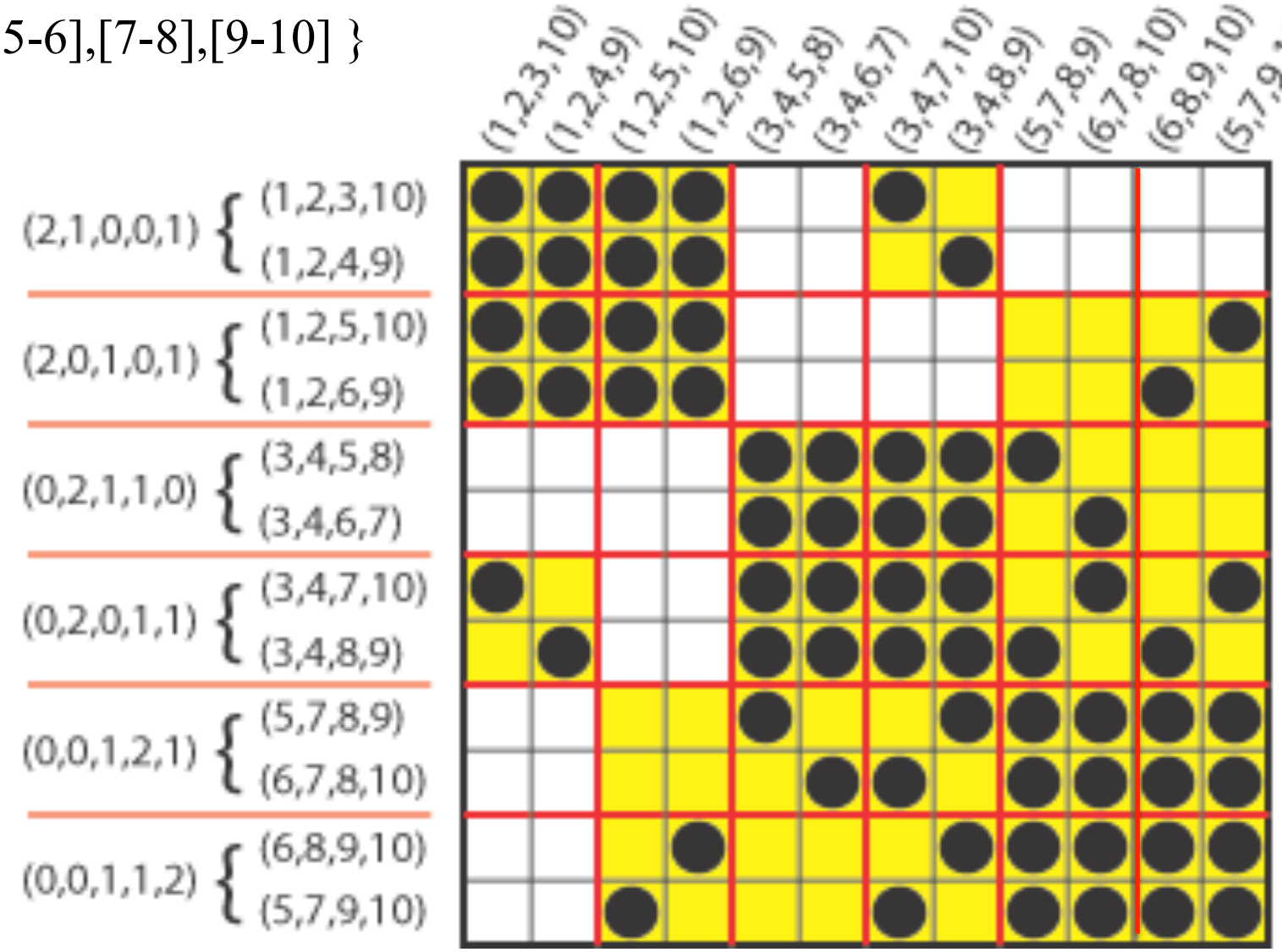
{ 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 10 , 11 , 12 } into
{ [1–4] , [5–8] , [9–12] }

Many-body states	*Cluster identifiers*
(2,3,4,7,9,12)	(3,1,2)
(1,2,4,7,8,12)	(3,2,1)
(1,4,5,7,8,9)	(2,3,1)
(1,2,9,10,11,12)	(2,0,4)

- Let S and T be cluster identifiers for many-body states s and t, respectively. If $|| S - T ||_1 > 4$, then $H_{s,t} = 0$.

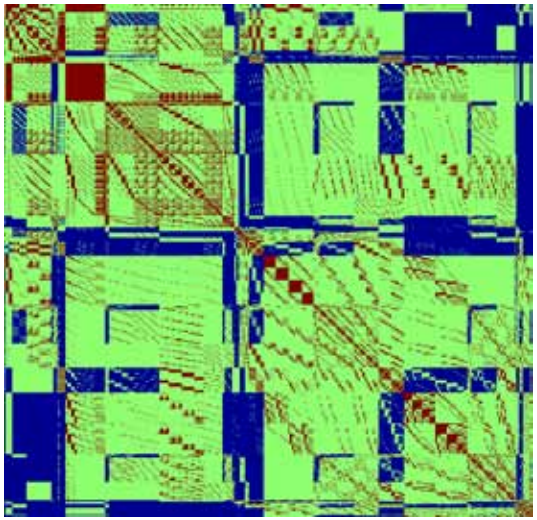
Tiny Example with Blocking

{ [1-2],[3-4],[5-6],[7-8],[9-10] }

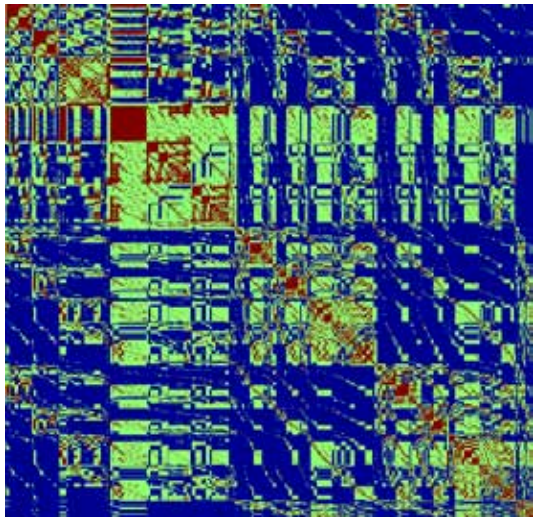


Coarse, Fine, and Multilevel Blocking

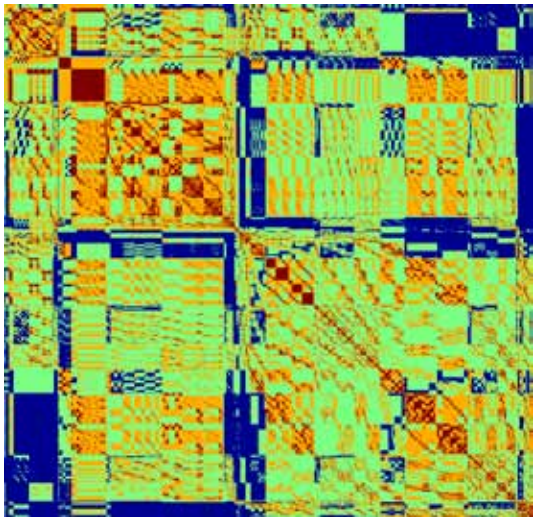
coarse



fine



fine on top of coarse



- Nonzero blocks
- Zero blocks
- Potentially nonzero blocks
- Potentially nonzero blocks in fine partitioning

^{6}He:

no blocking:	~ 43 minutes
one level:	180 seconds
multiple levels:	90 seconds

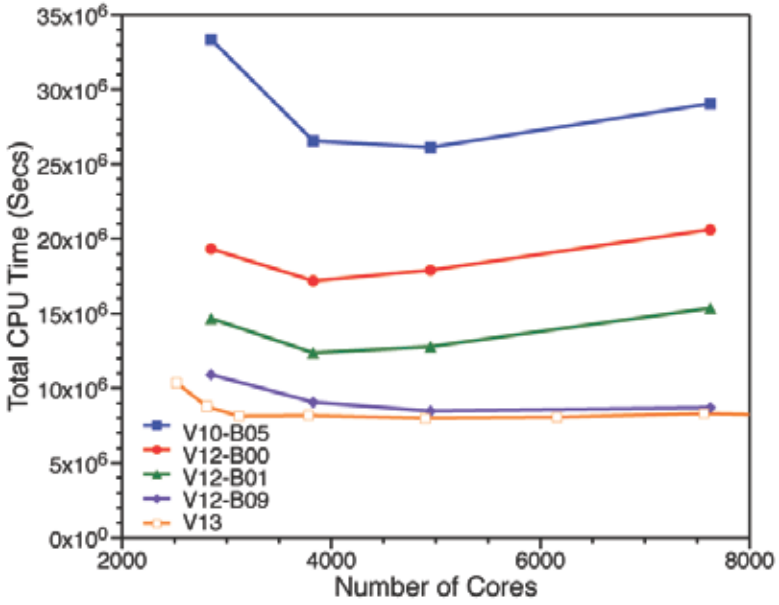
Performance Results

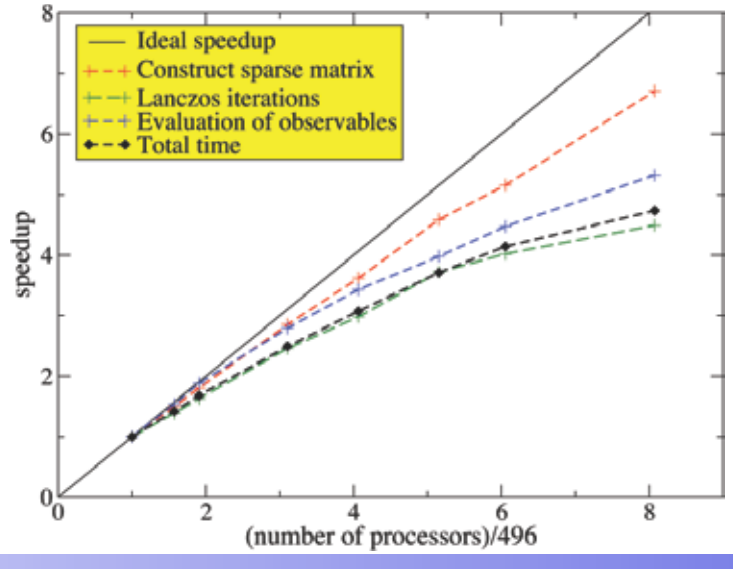
- Time to compute sparsity:

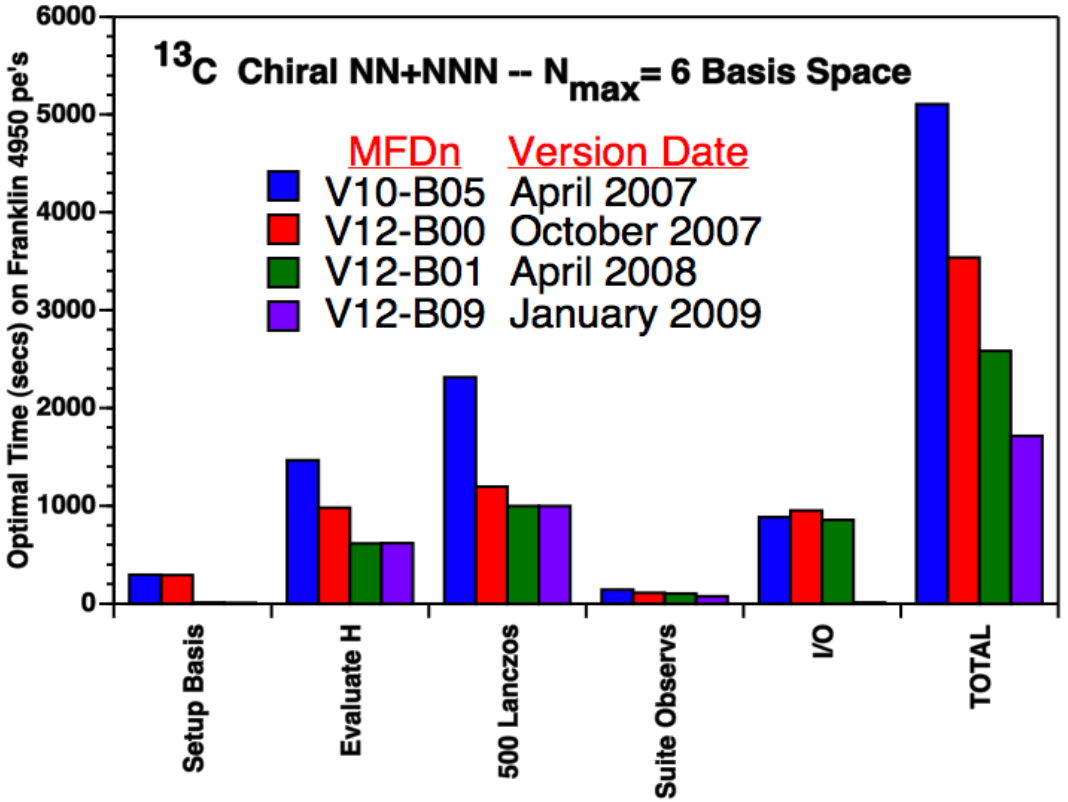
	No Blocking	*One Level*	*Multiple Levels*
^{6}He	~ 43 minutes	180 seconds	90 seconds
^{12}C	> 100 hours (?)	~ 1 hour	~ 13 minutes
^{16}O	> 100 hours (?)	~ 2 hours	~ 20 minutes

- In addition to reducing the time to determine the sparsity structure of *H*, the block structure has the potential of improving the performance of sparse matrix-vector multiplications.

Performance of Nuclear Structure Calculations



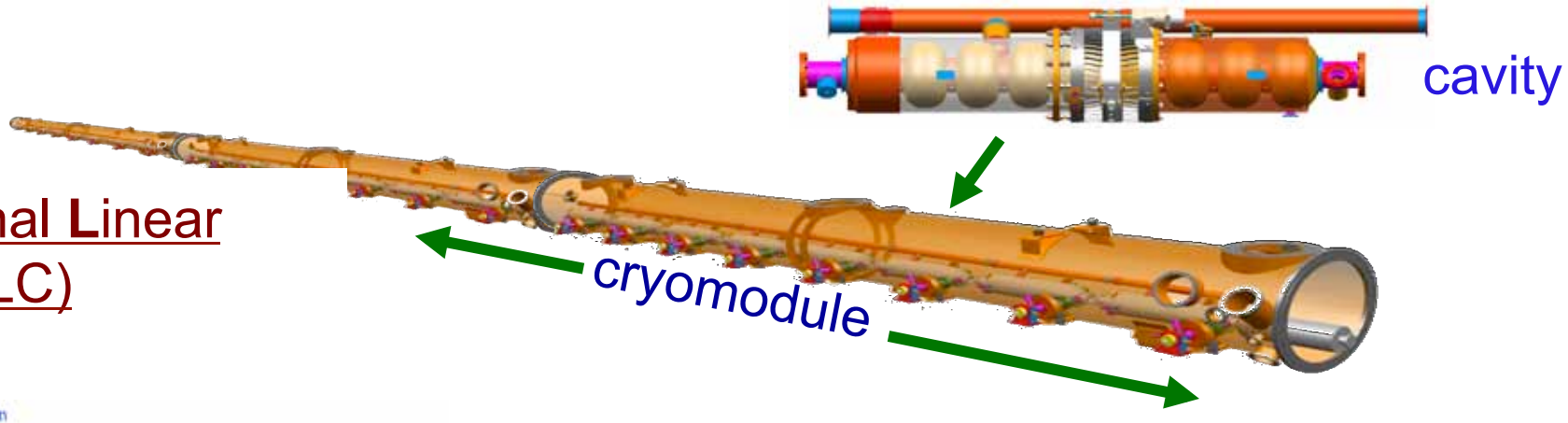




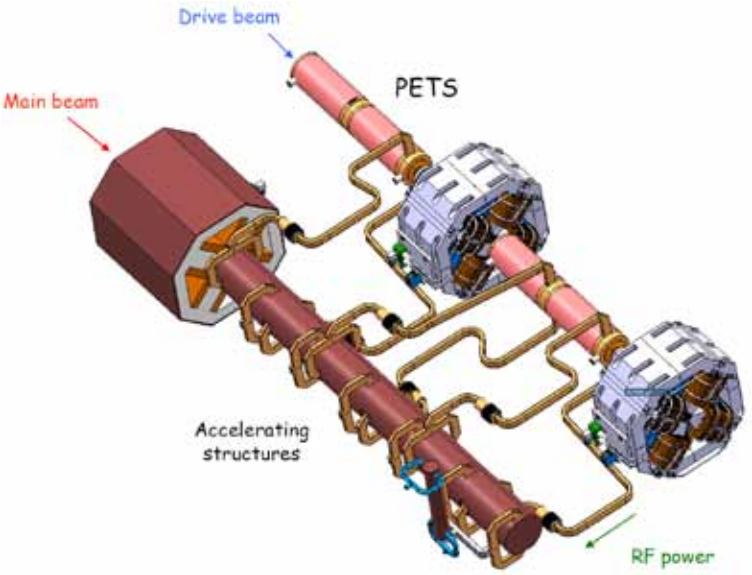
Dimension	38×10^6
# nonzero matrix elements	56×10^{10}
Input NNN matrix elements	3 Gbytes

High-Fidelity Modeling of Particle Accelerators

- Particle accelerators are important for high-energy and nuclear physics research.



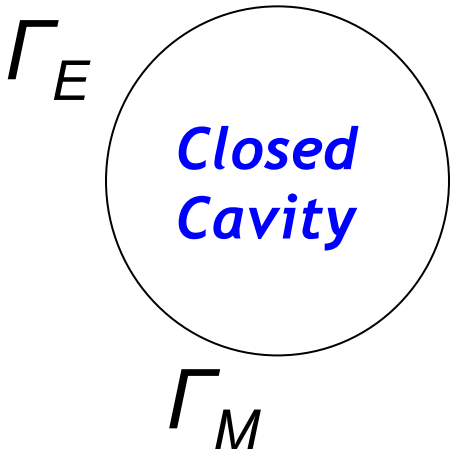
International Linear Collider (ILC)



Compact Linear Collider (CLIC) - *Two-Beam Accelerator*

Eigenvalue Problems for Cavity Modeling

Solving Maxwell's Equations in the 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 $\quad \mathbf{E} = \sum_i x_i \mathbf{N}_i$

=> find frequency and field vector of normal modes:

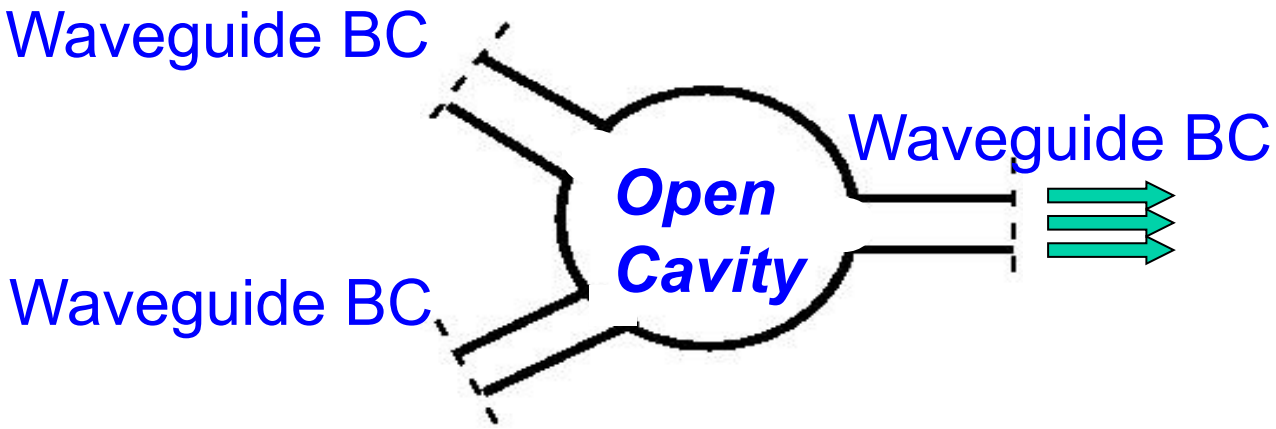
$$\mathbf{K}\mathbf{x} = k^2\mathbf{M}\mathbf{x}$$

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

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

Modeling Cavity Coupled to Multiple Waveguides

- Vector wave equation with waveguide boundary conditions leads to a ***complex non-linear eigenvalue problem***



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

where

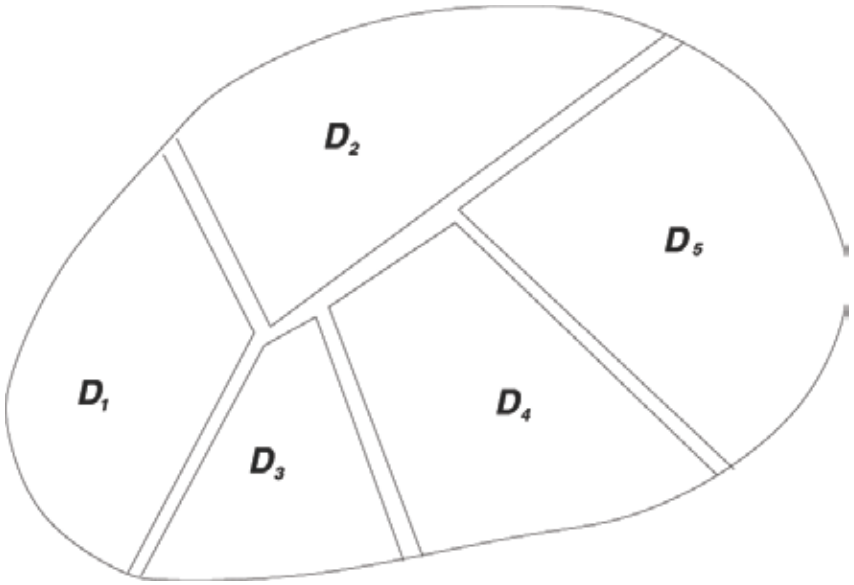
$$(\mathbf{W}_{mn}^{TE})_{ij} = \int_\Gamma \vec{\mathbf{e}}_{mn}^{TE}\cdot\mathbf{N}_i\,d\Gamma\int_\Gamma \vec{\mathbf{e}}_{mn}^{TE}\cdot\mathbf{N}_j\,d\Gamma$$
$$(\mathbf{W}_{mn}^{TM})_{ij} = \int_\Gamma \vec{\mathbf{e}}_{tmn}^{TM}\cdot\mathbf{N}_i\,d\Gamma\int_\Gamma \vec{\mathbf{e}}_{tmn}^{TM}\cdot\mathbf{N}_j\,d\Gamma$$

Solving the Eigenvalue Problems

- Gao, Husbands, Li, Ng, Yamazaki, Yang + Bai + Ko, Lee, Ng
- Linear Eigenvalue Problem (LEP)
  - Shift-and-Invert Lancos/Arnoldi
  - For shifted linear system
    - Sparse direct solvers (MUMPS, SuperLU, WSMP)
    - CG/GMRES with spectral multilevel preconditioner
- Quadratic Eigenvalue Problem (QEP)
  - NEP can be converted to a QEP for single waveguide mode
  - Second Order Arnoldi with Shift-and-Invert
- Nonlinear Eigenvalue Problem (NEP)
  - Nonlinear Jacobi-Davidson
  - Self consistent iterations

Sparse Linear Equations Solver

- Li, Yamazaki
- Compute a partitioning of the graph of A (using, e.g., PT-SCOTCH and ParMETIS).
  - The domains are balanced in size.
  - The separator is small.
  - Order the domains before the separator.



Block Factorization

$$\begin{bmatrix} D_1 & & & & & E_1 \\ & D_2 & & & & E_2 \\ & & D_3 & & & E_3 \\ & & & D_4 & & E_4 \\ & & & & D_5 & E_5 \\ F_1 & F_2 & F_3 & F_4 & F_5 & A_{\Gamma\Gamma} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix}$$

Block elimination

$\Rightarrow$

$$\begin{bmatrix} D_1 & & & & & E_1 \\ & D_2 & & & & E_2 \\ & & D_3 & & & E_3 \\ & & & D_4 & & E_4 \\ & & & & D_5 & E_5 \\ & & & & & S \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ y \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ c \end{bmatrix}$$

Schur Complement Method

- The diagonal blocks D_l can be eliminated in parallel.
  - Each diagonal block can be factorized either serially or in parallel, using, e.g., SuperLU, SuperLU_DIST, MUMPS, ...

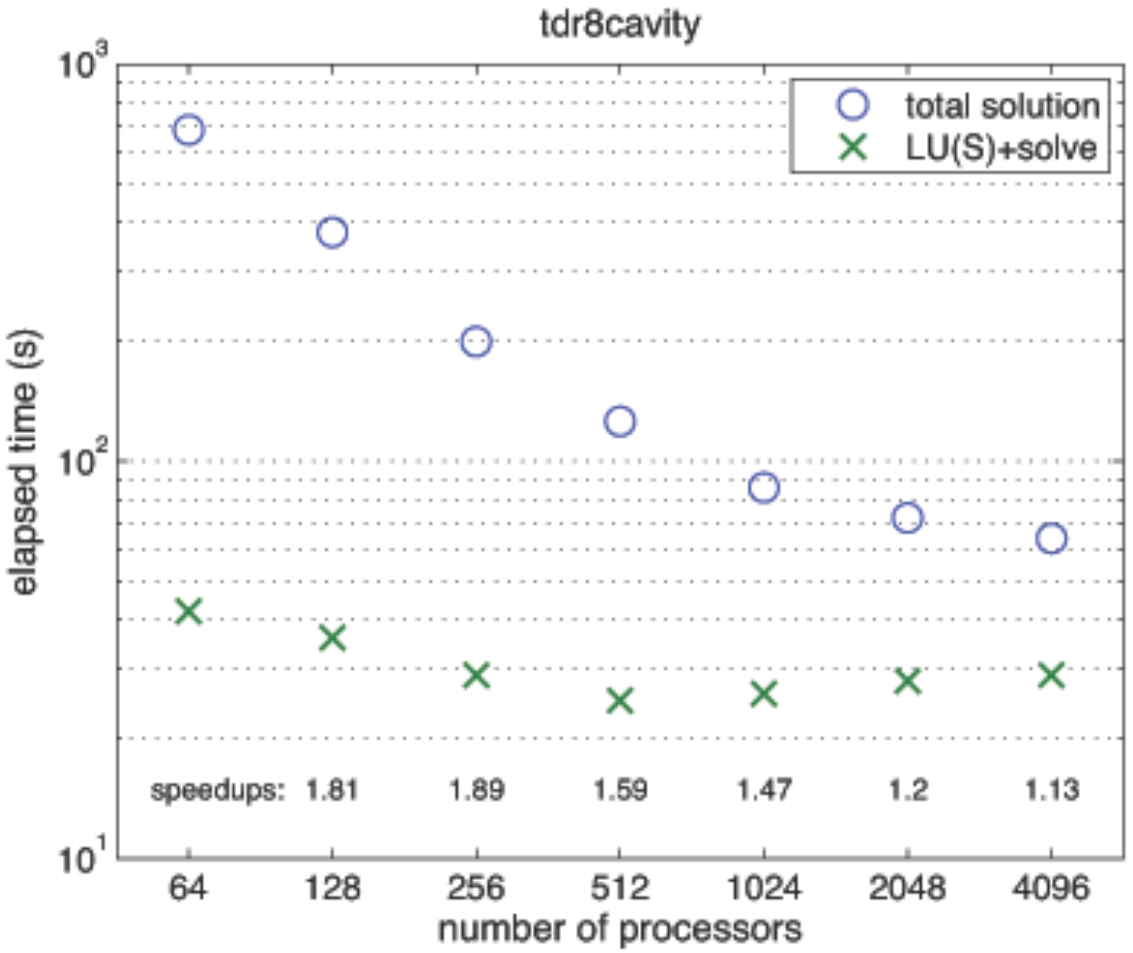
- Then the Schur complement is given by

$$S = A_{\Gamma\Gamma} - \sum F_l D_l^{-1} E_l$$

- The subsystem $Sy = c$ can be solved in a number of ways.

Hybrid Solver Performance

- ILC cavity problem:
  - Dimension = 17,799,228 (real symmetric, highly indefinite)
- Experimental setup:
  - PT-SCOTCH to extract 64 domains, each of size ~277K
  - SuperLU_DIST to factor each domain.
  - SuperLU_DIST to compute LU(S'), with S' ≈ S of size 57K, using 64 processors.
  - BICGStab from PETSc to solve Sy = c until rel residual < 10^{-12} (converged in ~10 iterations).



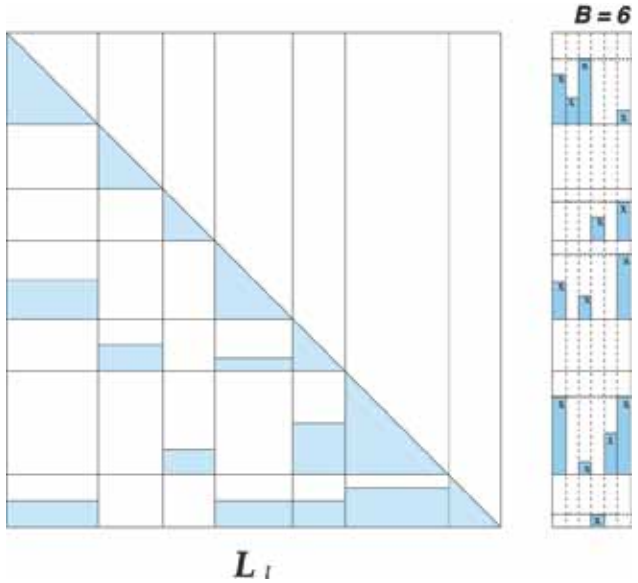
(on NERSC Cray XT-4)

Sparse Triangular Solution with Sparse RHS

- Suppose $D_l = L_l U_l$. Then

$$\begin{aligned} S &= A_{\Gamma\Gamma} - \sum F_l D_l^{-1} E_l \\ &= A_{\Gamma\Gamma} - \sum \left(U_l^{-1} F_l^T \right)^T \left(L_l^{-1} E_l \right) \\ &= A_{\Gamma\Gamma} - \sum W_l^T V_l \end{aligned}$$

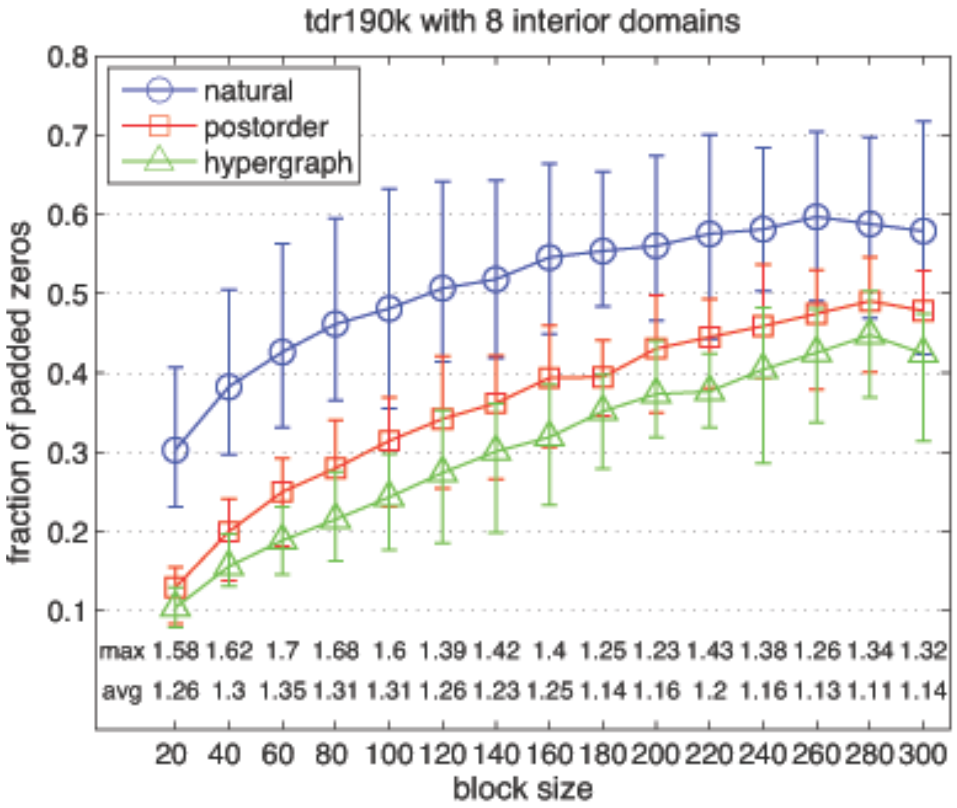
- Since E_l and F_l are generally sparse, we have to deal with the solution of sparse triangular systems with many sparse right-hand sides.
  - Both W_l and V_l may be sparse too.

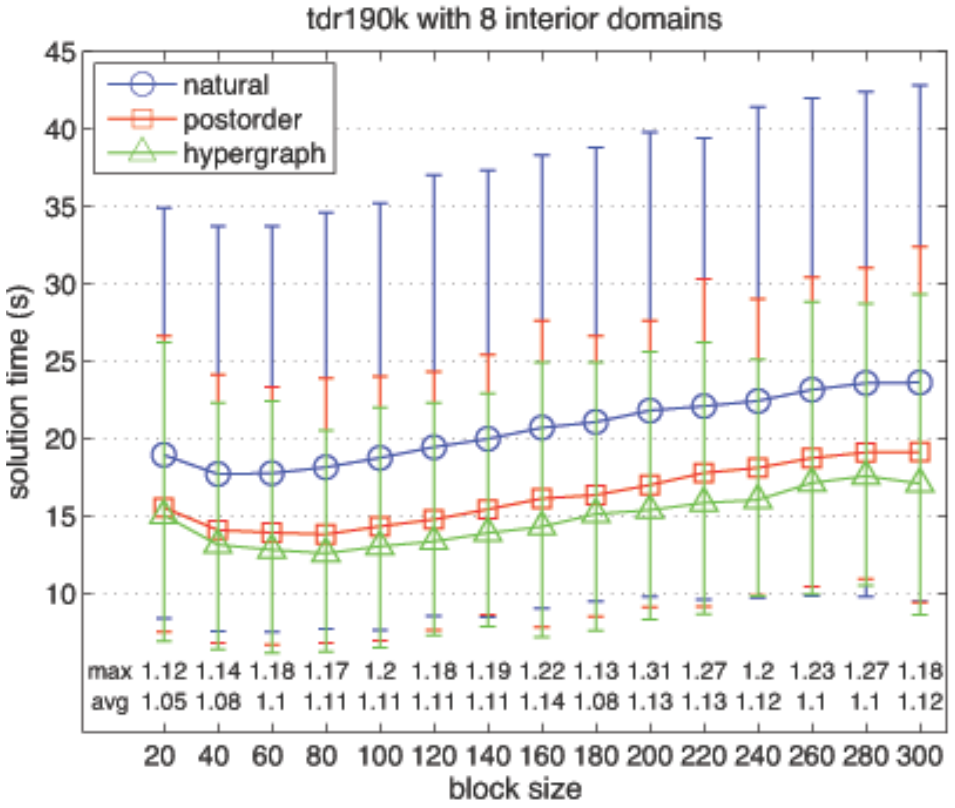


Sparse Triangular Solution with Sparse RHS

- Desirable to organize the computation so that
  - Sparsity of W_l and V_l is exploited,
  - Sparisty of L_l and U_l is exploited, and
  - Communication is optimized.
    - Sending empty messages (corresponding to zero blocks) is avoided.
  - Padding with zeros is minimized.
- Performance can be affected by ordering the right-hand sides.
  - Li, Yamazaki, Rouet, Uçar.
  - Order the right-hand sides according so that the row indices of the first nonzero entries are in ascending order.
  - Minimize the number of padded zeros by using a hypergraph model, which captures how the columns interact through their row structures.
    - A somewhat global view.

Sparse Triangular Solution with Sparse RHS





- Postorder greatly improves upon natural ordering.
- Hypergraph further improves upon postorder (though the improvements seem to be small).

Summary

- Much of the linear algebra work at DOE labs is driven by applications.
  - Stability, sparsity, dimension
- Algorithmic and software development is influenced by the hardware capabilities available.
- Only talked about nuclear physics and accelerator modeling.
  - But there are many other applications in which linear algebra plays a significant role …
    - Chemistry
    - Materials/nano sciece
    - Fusion
    - Environmental issues
    - …