Algebraic Sub-structuring (Domain Decomposition) for Large-scale Electromagnetic Application

Weiguo Gao   Chao Yang   Sherry Li
Lawrence Berkeley National Laboratory

Zhaojun Bai
University of California, Davis

Joint Work with Rich Lee, Kwok Ko (SLAC)

Outline

- Algebraic Multi-Level Sub-structuring (AMLS)
- Why does it work?
- Implementation issues
- Numerical results and performance
  - Focus on electromagnetic application
  - Compare with shift-and-invert Lanczos (SIL)
    - time, memory, accuracy
- DOE SciDAC Projects
  - Terascale Optimal PDE Simulations (David Keyes)
  - Advanced Computing for 21st Century Accelerator Science and Technology (Kwok Ko, Rob Ryne)

Background

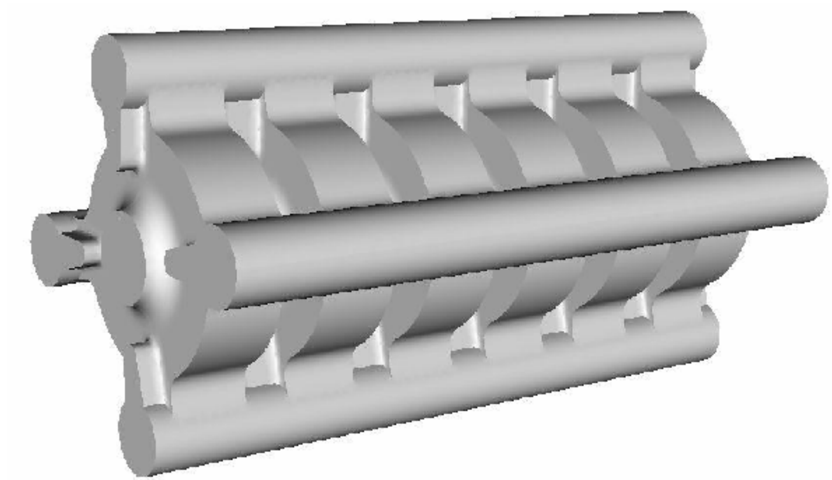
- Generalized sparse eigenvalue problem: $K\,x = \lambda\,M\,x$
  - K symmetric, M SPD
  - Need large number of small nonzero eigenvalues

- Sub-structuring dates back to the 1960's (CMS)
  - Plenty of engineering literature
- AMLS has recently been used successfully in structural engineering (Bennighof, Kaplan, Lehoucq)
  - Compute vibration modes
  - Perform frequency response analysis

- Open questions remain as a general-purpose solver
  - Accuracy
  - Performance

EM Application

- Can we extend the success story from structural engineering to electromagnetic applications (accelerator cavity design)?
- Important for the next generation linear accelerator design (SLAC)



- Curl-curl formulation of Maxwell's equation

$$\nabla \times (\nabla \times \mathbf{E}) - \lambda \mathbf{E} = 0 \;\; \text{in } \Omega$$

$$n \times \mathbf{E} = 0 \;\; \text{on } \Gamma_{\mathbf{E}}$$

$$n \times (\nabla \times \mathbf{E}) = 0 \;\; \text{on } \Gamma_{\mathbf{B}}$$

Challenges of Eigenproblems in Accelerator Design

- Large matrix size for realistic structures
  - Tens of millions to hundreds of millions

- Small eigenvalues (tightly-clustered) out of a large-eigenvalue dominated eigenspectrum
  - Many small nonzero eigenvalues desired

- Large null space in the stiffness matrix
  - Up to a quarter of the dimension

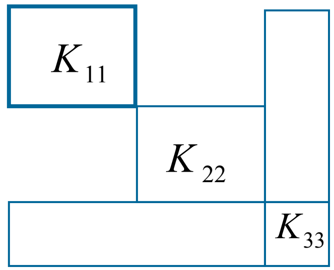
- Requires high accuracy for eigenpairs

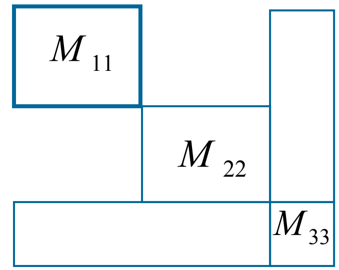
To Answer The Question…

- Look at the description of the algorithm to see if it is applicable
- Analyze approximation properties of the algorithm (error estimate)
- Examine the complexity of the implementation

Single Level Substructuring

1. Partitioning & reordering for (K,M)

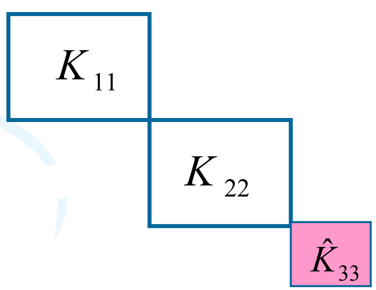


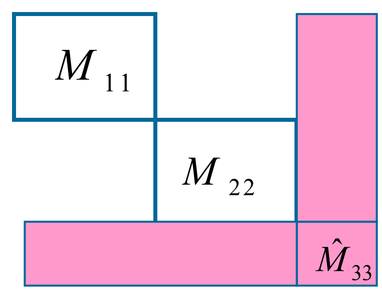


2. Block Gaussian elimination (congruence transformation)

$$\hat{K} = L^{-1} K L^{-T}$$

$$\hat{M} = L^{-1} M L^{-T}$$





Single Level (cont)

3. Sub-structure calculation for a subset of the modes (mode selection)

$$K_{ii} v^{(i)} = \mu^{(i)} M_{ii} v^{(i)}, \quad i = 1,2$$

$$\hat{K}_{33} v^{(3)} = \mu^{(3)} \hat{M}_{33} v^{(3)}$$

3. Subspace assembling

$$S = \begin{pmatrix} S_1 & & \\ & S_2 & \\ & & S_3 \end{pmatrix} \qquad S_i = \left(v_1^{(i)} \ v_2^{(i)} \ \cdots \ v_{k_i}^{(i)} \right), \quad i = 1, 2, 3$$

Single Level (cont)

5. Projection (Rayleigh-Ritz)

$$\left(S^T \hat{K} S\right) q = \theta (S^T \hat{M} S) q$$

6. Unravel

$$D = \mathrm{diag}\left(\theta_1, \theta_2, \cdots, \theta_m\right)$$

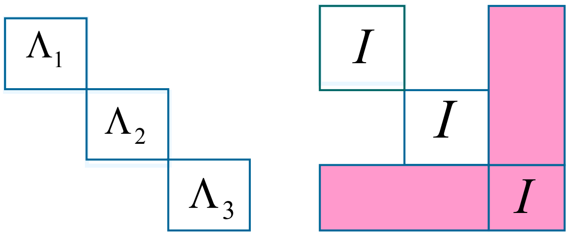
$$Z = L^{-T} S Q_m$$

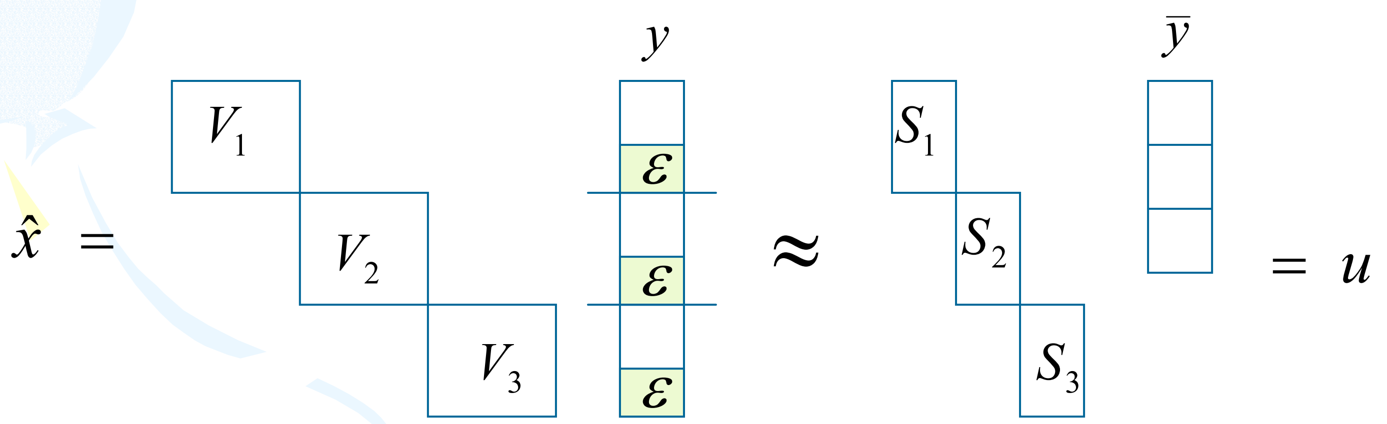
$$(\; Q_m = (q_1, q_2, \cdots, q_m) \, , \, \hat{K} = L^{-1} K \, L^{-T} \;)$$

Algebraic Analysis

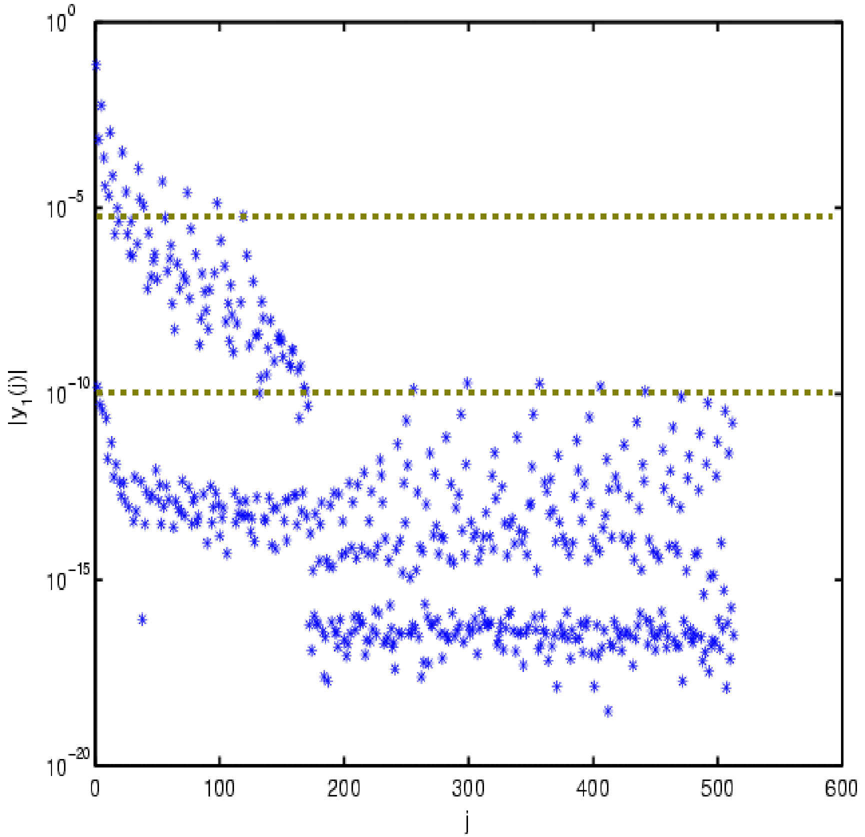
$$\hat{x} = \begin{pmatrix} V_1 & & \\ & V_2 & \\ & & V_3 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}$$

Canonical form



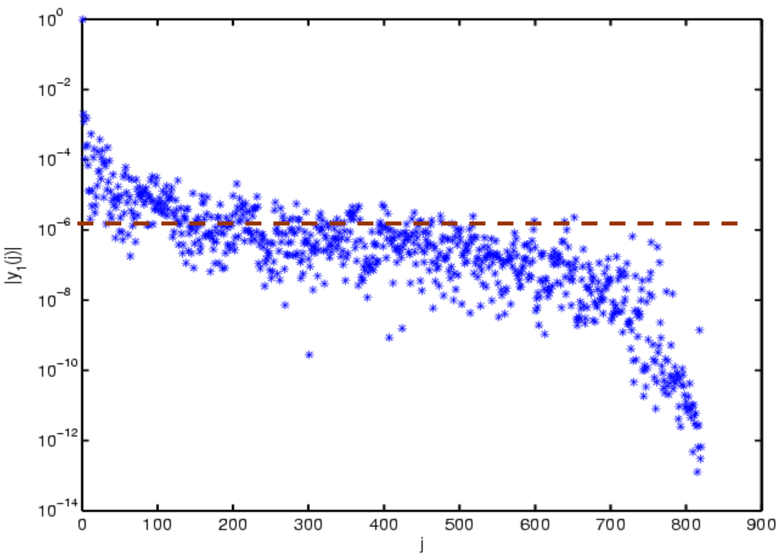


Example 1 (BCS09)

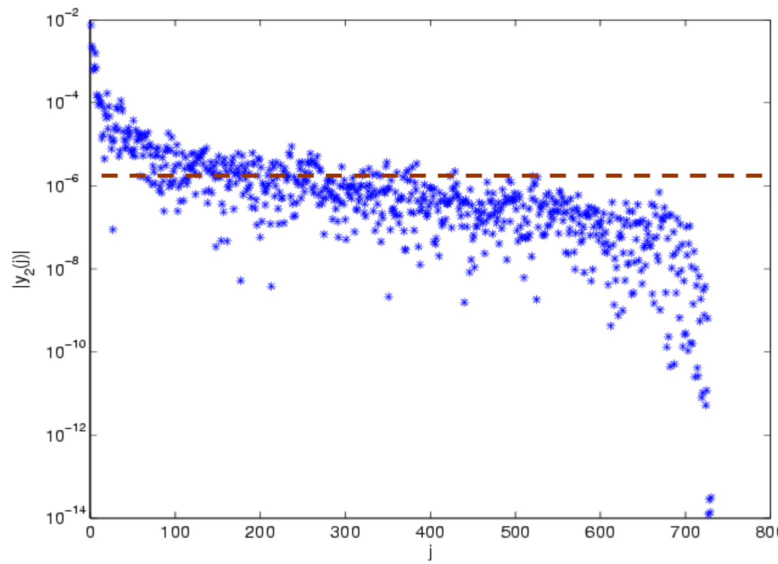


y_1 - plot

Example 2 (Accelerator Model)



y_1 - plot



y_2 - plot

Error Bound

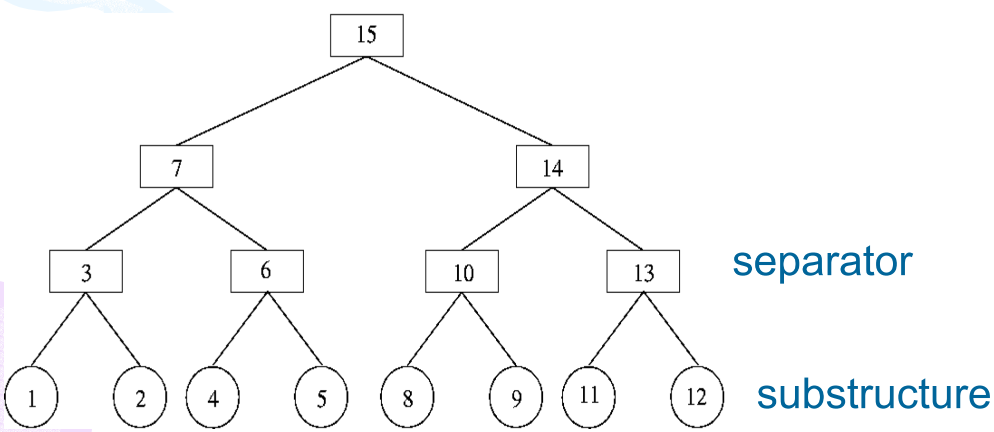
$$\theta_1 - \lambda_1 \le (\lambda_n - \lambda_1)\, h^2$$

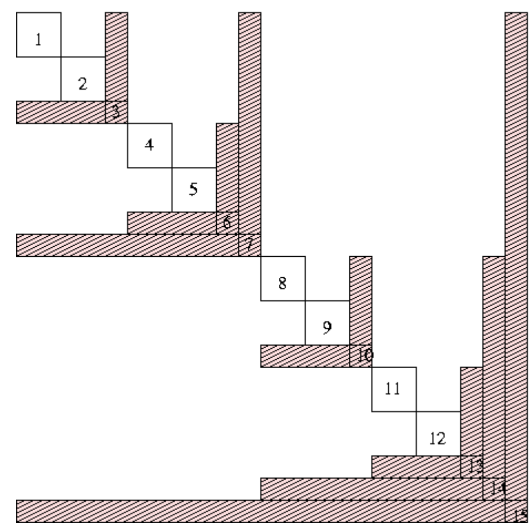
$$\sin \angle_{\hat{M}} (u_1, \hat{x}_1) \le \sqrt{\frac{\lambda_n - \lambda_1}{\lambda_2 - \lambda_1}}\, h$$

- h measures the size of the "truncated" components of y
- Related work
  - Bourquin et al. (CMS analysis)
  - Bekas & Saad (Spectral Schur Complement)
  - Elssel & Voss (minmax theory for rational eigenvalue problem)

Multilevel Algorithm (AMLS)

1. Matrix partitioning and reordering using nested dissection
2. Block elimination and congruence transformation
3. Mode selection for sub-structures and separators
4. Subspace assembling
5. Projection calculation
6. Eigenvalues of the projected problem



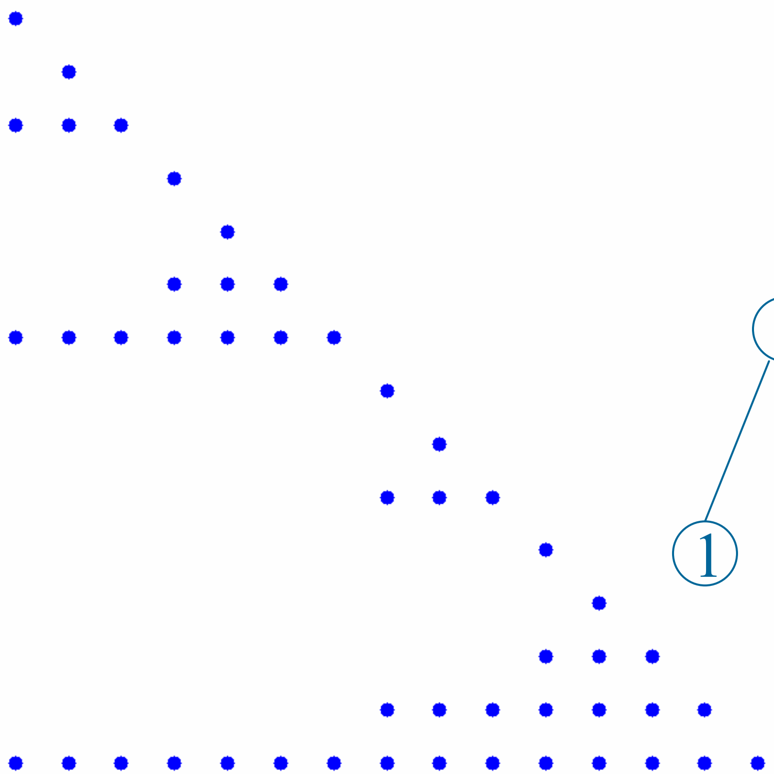


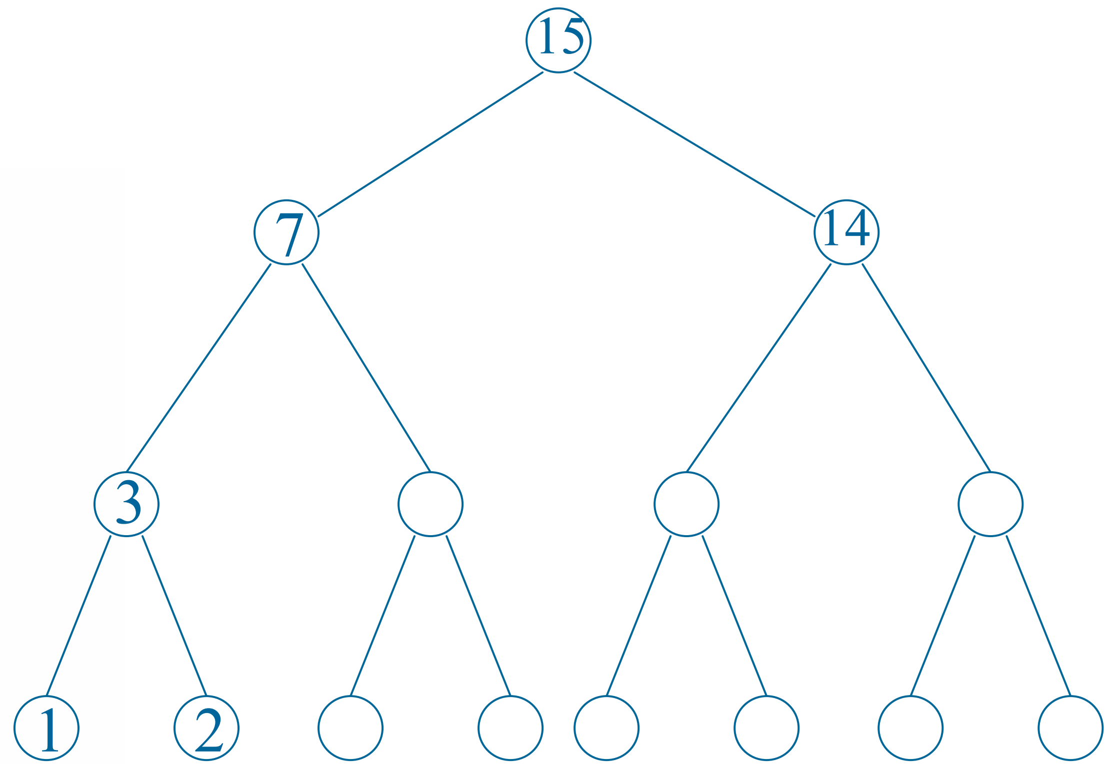
Implementation

- Major Operations:
  - Transformations and projection
    - Steps 2-5 can be interleaved
  - Eigenpairs of the projected problem

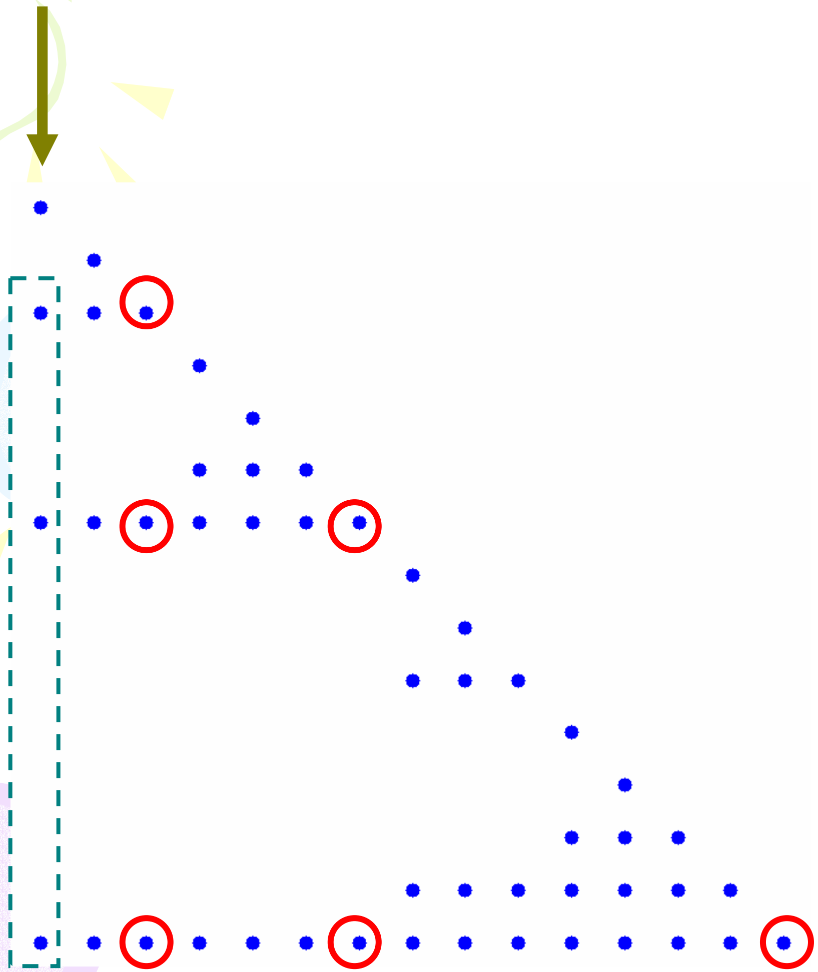
- Cost:
  - Flops: more than a single sparse Cholesky factorization
  - Storage: Block Cholesky factor + Projected matrix + some other stuff
  - NO Triangular solves, NO orthogonalization

Task Dependency (Greedy Algorithm)



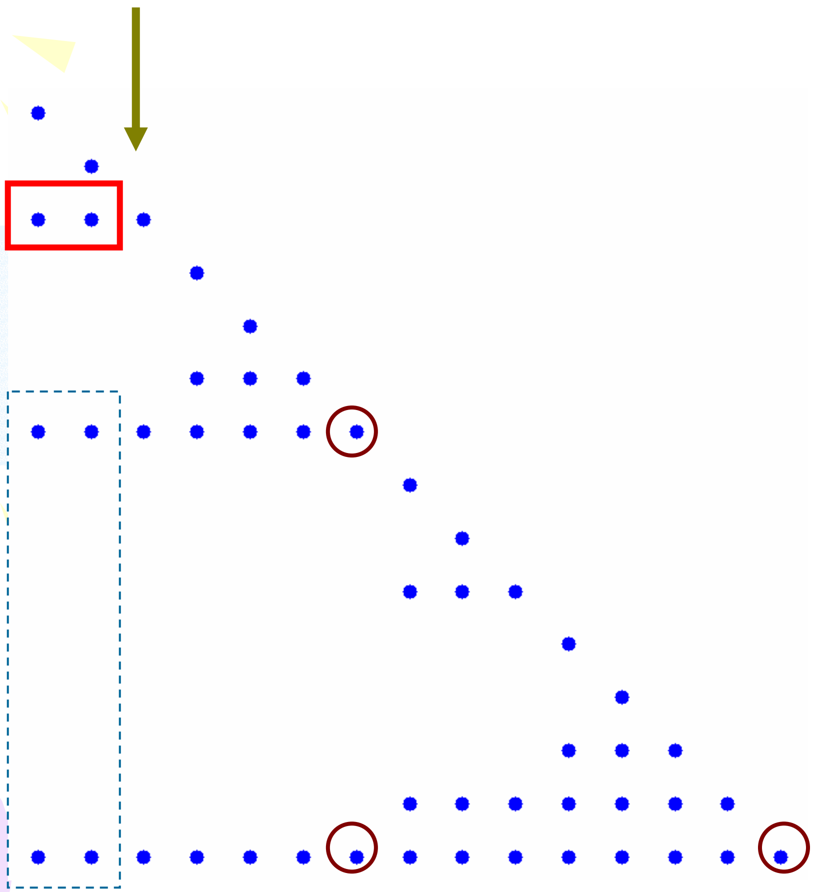


Bottom Level



- Eliminator: L_{i1}
- Modes: $K_{11}V_1 = M_{11}V_1\Lambda_1$
- Eliminate *K*: one sided update
- Congruence transformation on *M*: two sided update
- Store half-projected $\hat{M}_{i,1}$ (dashed box)

Higher Level



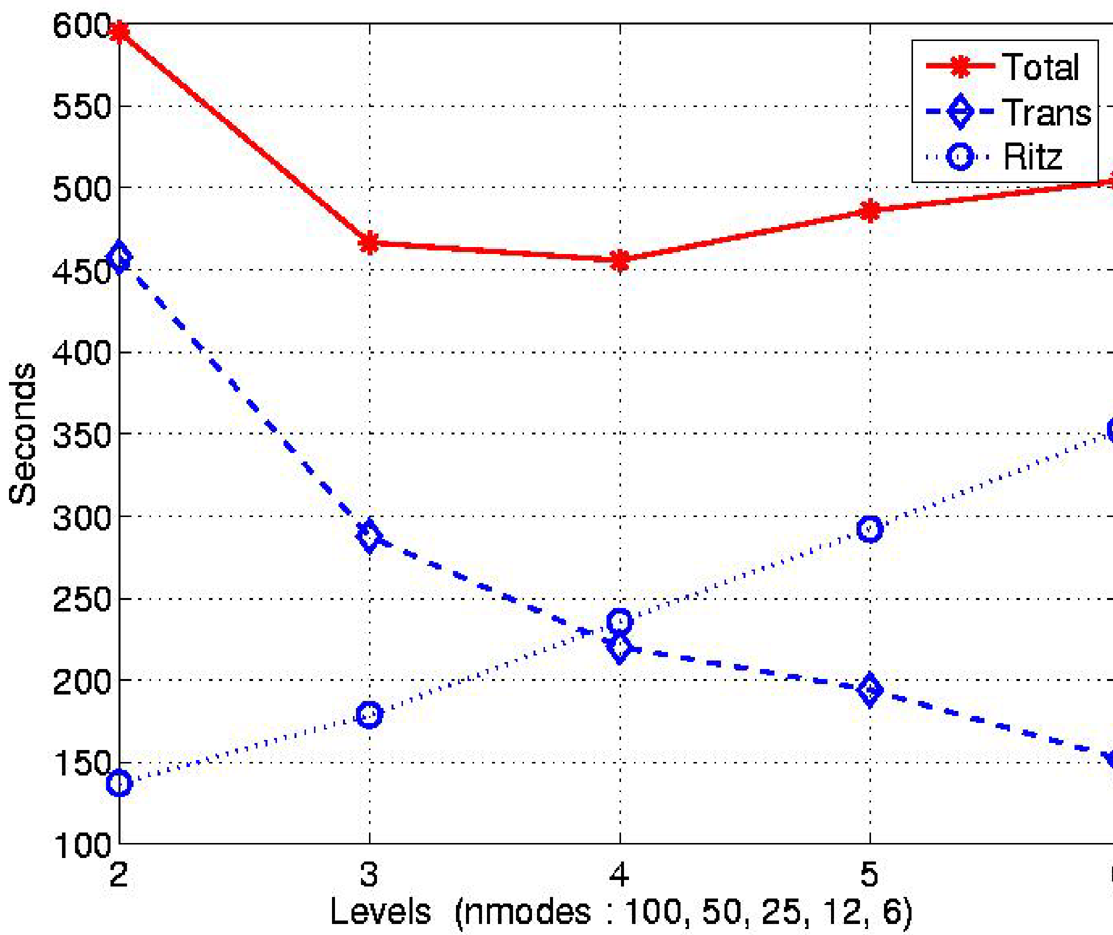
- Additional updates of previously “half-projected” columns

  (dashed box)

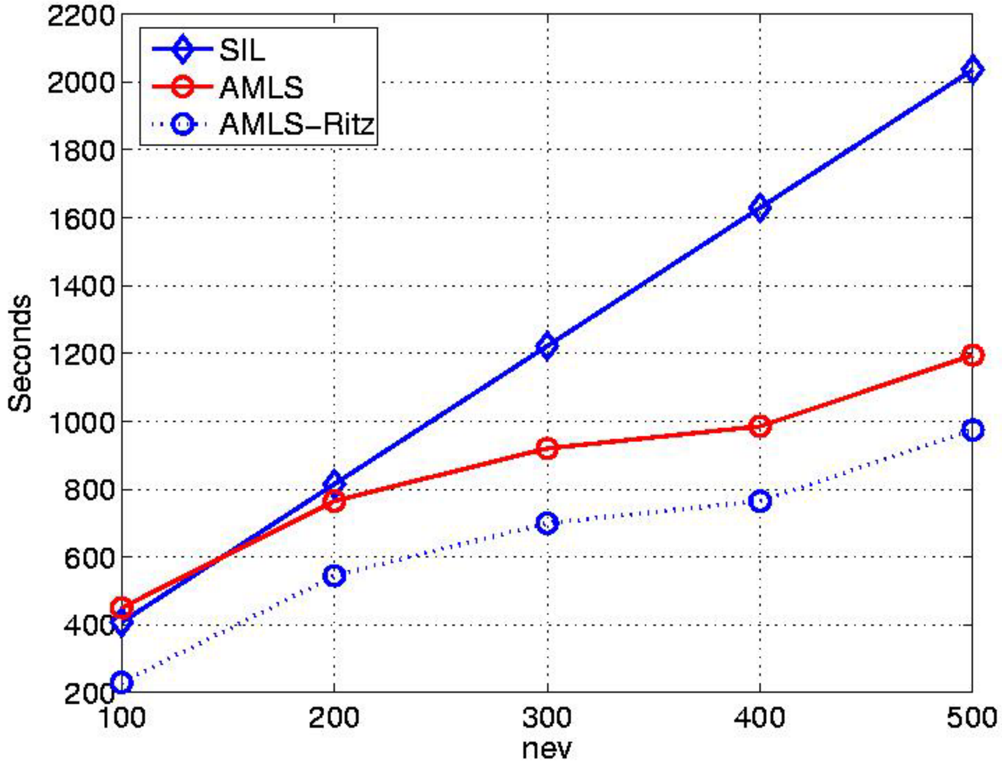
- Completion of some blocks (red box)

Example: Accelerator Model

- A 6-cell DDS structure
- N = 65K
  NNZ = 1455772
  nev = 100
- All the coupling modes are selected
- SIL took 407 sec (ARPACK + sparse LDL^T)

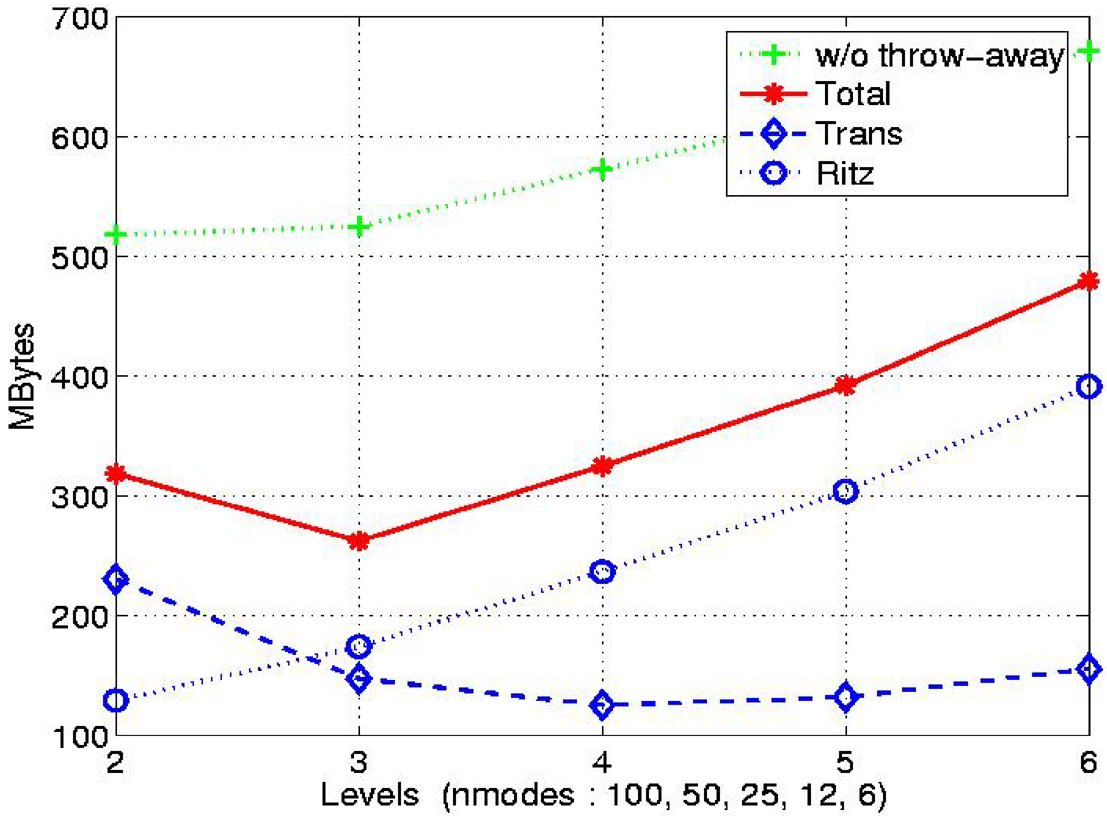


When Is AMLS Faster?



- Many eigenvalues are wanted (up to ~8% in this case)
- SIL requires multiple shifts (factorizations)

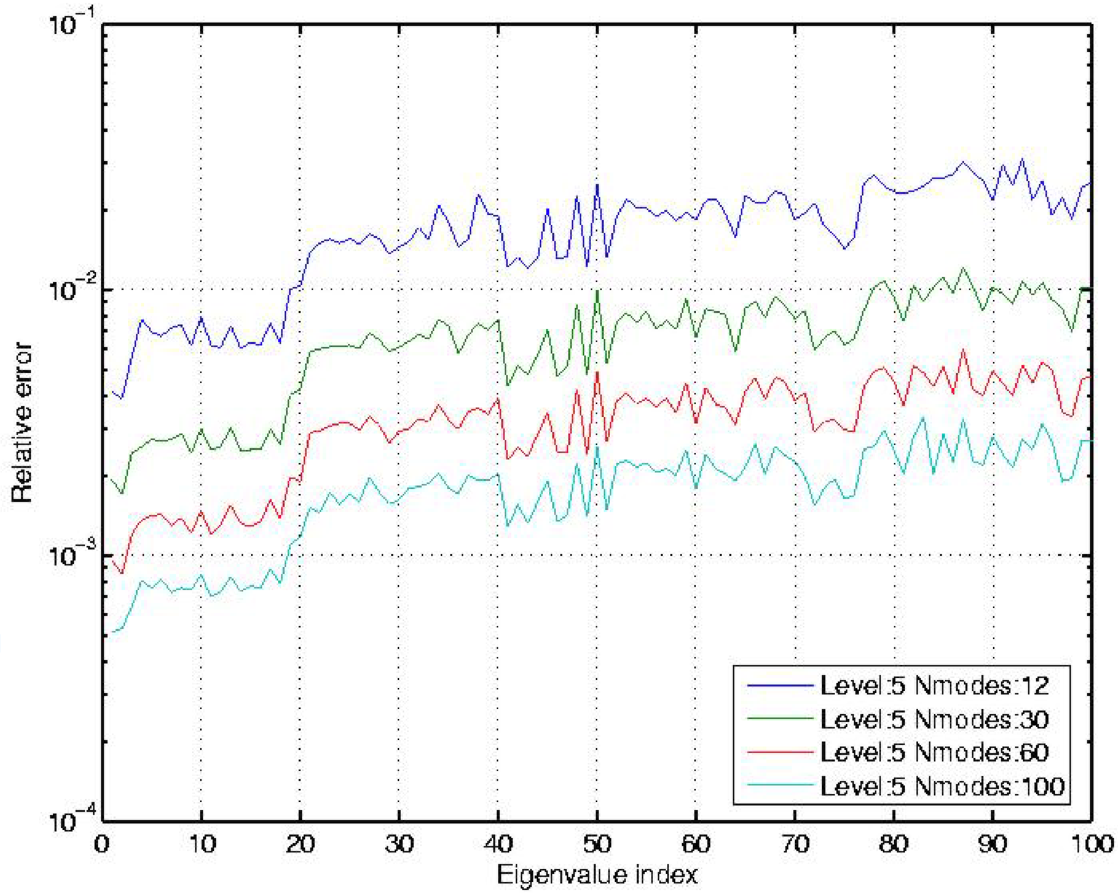
Memory Profile



- Save up to 50% memory with 13% re-compute time
- SIL needs ~308 Mbytes memory

Accuracy Compared with SIL

- Levels=5, increasing nmodes of sub-structures



Concluding Remarks

- EM in accelerator simulation is a truly challenging engineering problem
- Better understanding of accuracy
- AMLS software to be released
  - General-purpose, memory efficient
  - Application-tuned: null space handling
- Performance advantage shows up when:
  - The problem is large enough
  - A large number of eigenpairs are needed
- Many tuning parameters
  - Number of levels, number of modes, tolerances