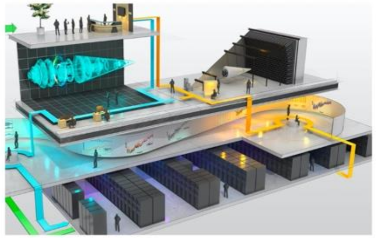
Multi-Region Solver and Algebraically-Transformed Block-Diagonal Preconditioner for High-Q Cavity Problems



Vinh Dang, Robert Pfeiffer, and Joseph Kotulski (1324)

{vqdang, rapfeif, jdkotul}@sandia.gov

Trilinos User Group Meeting 2024

October 22 – 24

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

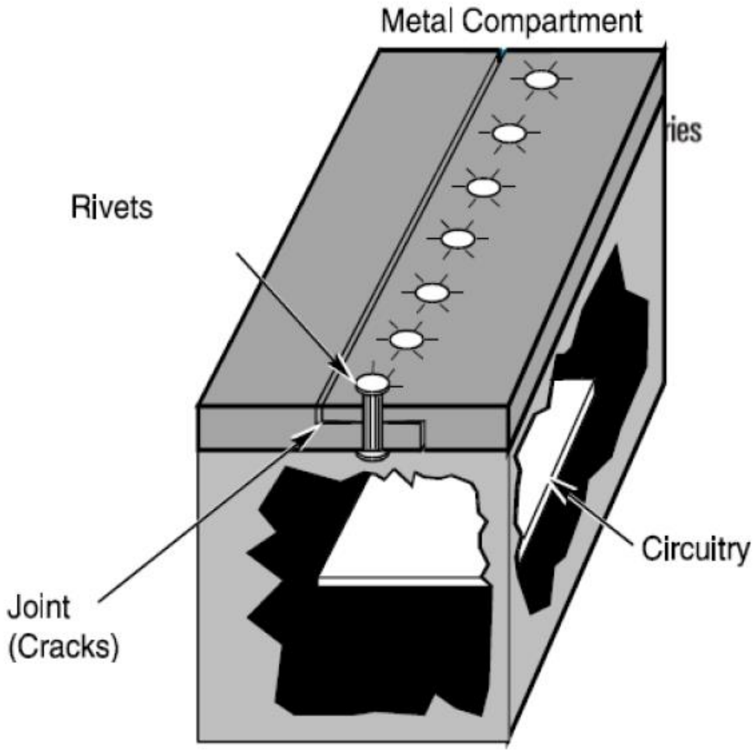
SAND2024-14180C

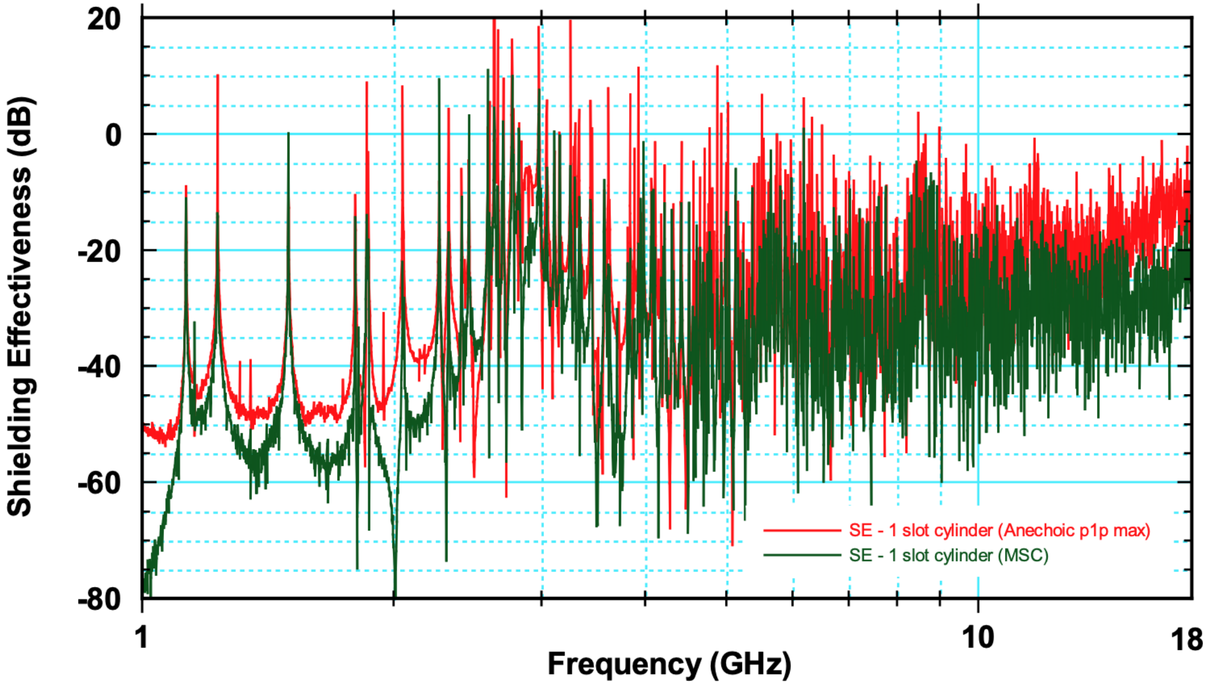
Outline

- Overview of Electromagnetic Coupling Problem
- Solution Technique
- Multi-Region Solver and Preconditioners
- Preliminary Results
- Conclusions and Future Work

Electromagnetic Coupling Problem

Electromagnetic (EM) energy couples into systems in many different ways. Our focus is energy coupling through mechanical seams or joints in the system housing, which acts as a good but imperfect EM shield. The joints form slots that allow EM energy into the system. Therefore, the problem has three regions that must be well-characterized: the slots, the cavity exterior and the cavity interior





Shielding effectiveness: $SE(dB) = 20\ log_{10}\left(\frac{E_{interior}}{E_{exterior}}\right)$

[1] Salvatore Campione et al., *Modeling and Experiments of High-Quality Factor Cavity Shielding Effectiveness*, 2019 International Applied Computational Electromagnetics Society Symposium (ACES), Miami, FL, USA, 2019.
[2] Matthew B. Higgins and Dawna R. Charley, *Electromagnetic Radiation (EMR) Coupling to Complex Systems: Aperture Coupling into Canonical Cavities in Reverberant and Anechoic Environments and Model Validation,* SAND2007-7931

Solution Technique

Gemma - Linear Electromagnetic Code

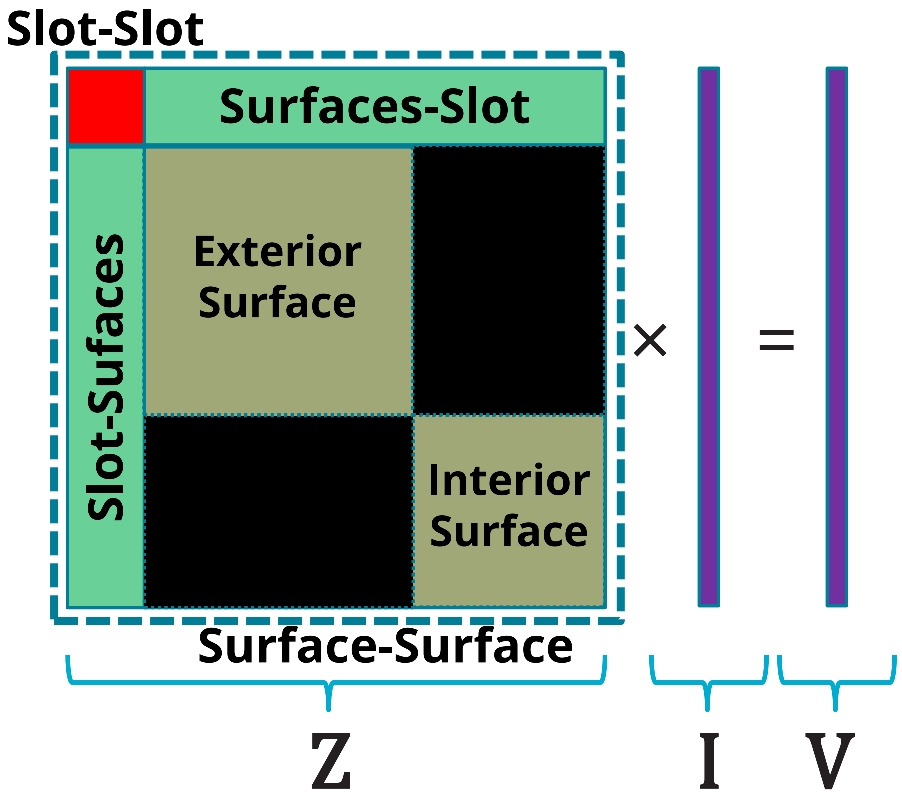
- A hybrid frequency-domain EM code (based on (method of moments - MoM), key features:
  - Narrow slot sub-cell model to characterize slots and capture EM penetration through the slots
  - Surface integral equation methods for the exterior surface and the interior surface of cavity:

$$L\{\mathbf{J_S}\} = \frac{1}{j\omega\mu}\hat{\mathbf{n}} \times \mathbf{E_{inc}}$$

Expand unknown and test the integral equation using the same set of basis functions:

$$\mathbf{J_S}(\mathbf{r}) \approx \sum_n I_n \mathbf{f_n}(\mathbf{r})$$

$$\int_S \mathbf{f_m} \cdot L\{\mathbf{J_S}\}\, ds = \frac{1}{j\omega\mu}\int_S \mathbf{f_m} \cdot (\hat{\mathbf{n}} \times \mathbf{E_{inc}})\, ds$$



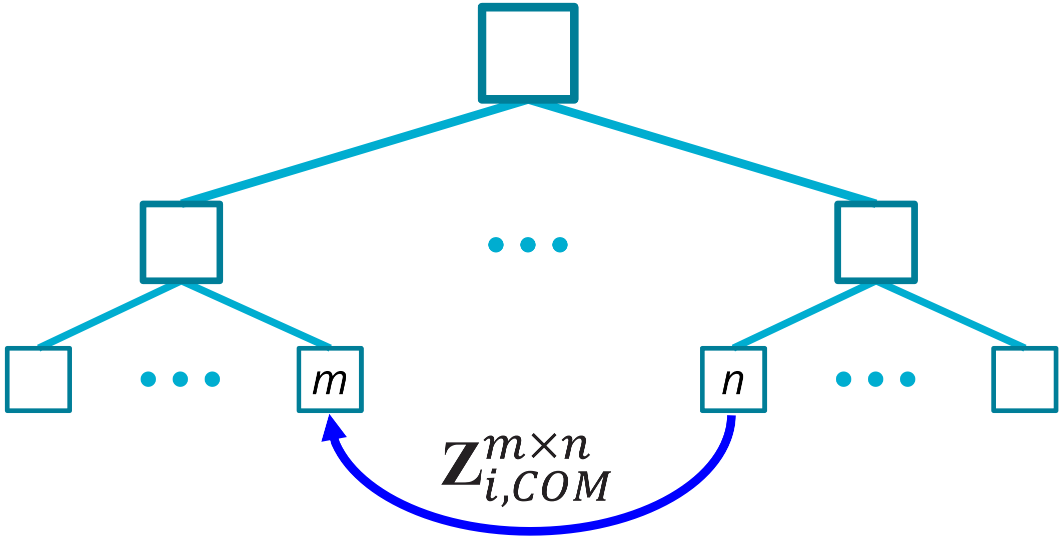
- Dense, complex linear system, ZI = V, solved by:
  - Direct solution, such as ADELUS ($O(N^3)$ complexity)
  - Iterative solution, such as GCR or GMRES ($O(N^2)$ complexity)
  - Iterative solution using fast numerical techniques + GCR or GMRES ($O(N \log N)$ complexity)
  - Due to the matrix condition number an appropriate preconditioner is needed

Fast Numerical Technique: Adaptive Cross Approximation (ACA)

- Main ideas: (i) use oct-tree or binary-tree to partition and group *N* unknowns to *P* boxes; (ii) use conventional MoM technique for near interactions and ACA for far interactions:

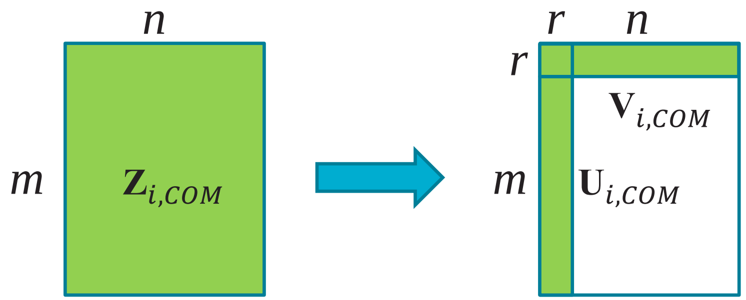
$$\mathbf{Z}\cdot\mathbf{I}=\mathbf{V} \Rightarrow \mathbf{Z}_{near}\cdot\mathbf{I}+\mathbf{Z}_{far}\cdot\mathbf{I}=\mathbf{V} \text{ or } \mathbf{Z}_{MOM}\cdot\mathbf{I}+\mathbf{Z}_{COM}\cdot\mathbf{I}=\mathbf{V}$$

  - 2 boxes interact to form a matrix block
  - Exploiting the rank deficiency of far matrix blocks (i.e. low-rank approximation) which is done on-the-fly (i.e. compressed matrix blocks never fully populated). Reduction of the memory footprint for the matrix
  - Note: since the full matrix is not stored an iterative solver is required



Oct-tree or binary-tree partitioning scheme

$$\mathbf{Z}_{i,COM}^{m\times n}\approx\tilde{\mathbf{Z}}_{i,COM}^{m\times n}=\mathbf{U}_{i,COM}^{m\times r}\cdot\mathbf{V}_{i,COM}^{r\times n}$$



Far interactions
(ACA – low-rank approximation of far blocks)

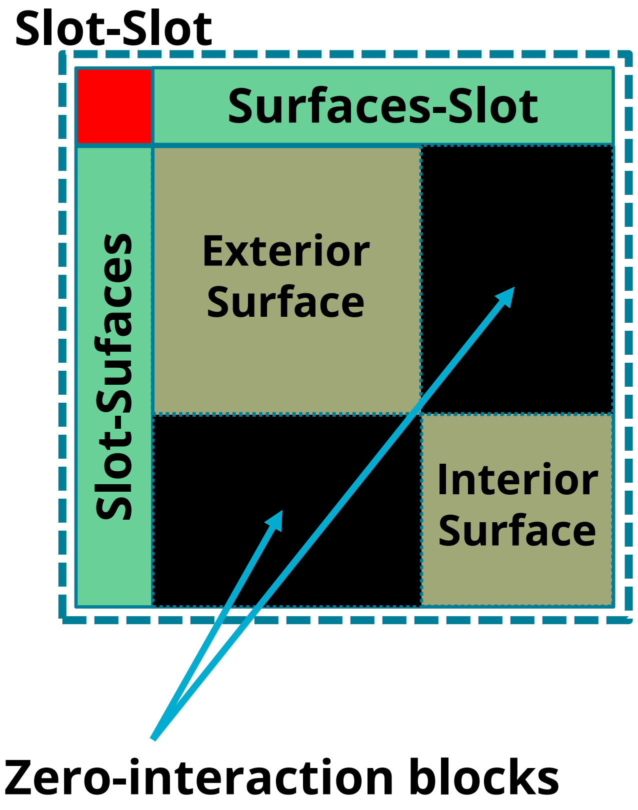
MOM: uncompressed method-of-moments matrix blocks (near interactions)
COM: compressed method-of-moments matrix blocks (far interactions)

K. Zhao et.al., "The adaptive cross approximation algorithm for accelerated method of moments computations of EMC problems," in *IEEE Transactions on Electromagnetic Compatibility*, vol. 47, no. 4, pp. 763-773, Nov. 2005

Multi-Region Solver and Preconditioners

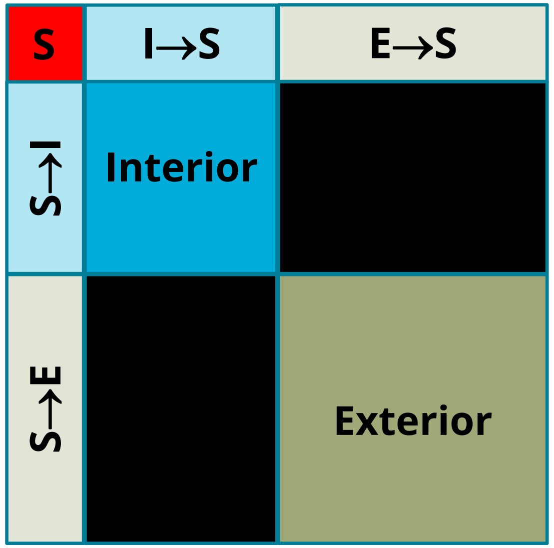
Multi-Region Solver Framework

- A multi-region ACA-accelerated solver framework for high-Q resonant cavity problems on distributed-memory hardware-accelerated systems
  - Choose appropriate solver for each region interaction
    - Enable Integration of the Fast Multipole Method in the framework as our future work
    - Extensible to modeling general complex, multiscale structures
  - Target performance portability → providing a path to run on upcoming next generation architectures
    - Kokkos and Kokkos Kernels
  - Use Trilinos iterative solvers



Multi-Region Solver Implementation

- Decompose the original problem into regions and assign appropriate solvers to different region interactions
  - ACA solver or MoM solver: the interior region and the exterior region having their own trees
  - Slot solver (no approximation): other region interactions
- A table to describe the interactions (and solver types) among regions
- Multi-region solver loops through the *domain_descriptor* table and call the right sub-solver's {*constructor*, *multiplyMatrixVector*, *constructPreconditioner*, *applyPreconditioner*} according to its label



S: Slot Self Interaction
I→S: Interior→Slot Interaction
E→S: Exterior→Slot Interaction
S→I: Slot→ Interior Interaction
S→E: Slot→ Exterior Interaction

Slot Solvers

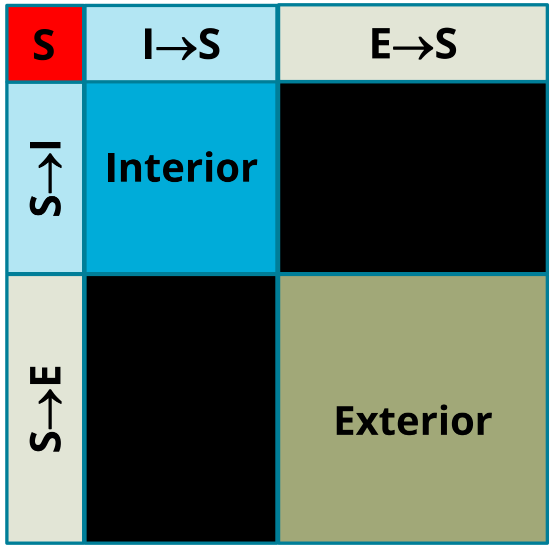
"0": no interaction
"1" : Slot solver
"2": MoM solver
"3": ACA solver

1	1	1
1	2	0
1	0	3

Workload Distribution onto MPI Ranks

Workload distribution is done for each region

- Slot solver: matrix blocks (that comprise slot-related interactions) are block-based distributed to MPI ranks
- MoM solver: matrix blocks are distributed to distributed tree nodes on MPI ranks (referred to [1] for details)
- ACA solver:
  - Matrix blocks are partitioned onto MPI ranks for load balancing via block indexing
    - No MPI rank can have more or less than one block than any other
  - Distribution scheme is applied separately to MOM blocks and COM blocks
    - MPI ranks have both MOM and COM blocks



MOM: uncompressed method-of-moments matrix blocks (near interactions)
COM: compressed method-of-moments matrix blocks (far interactions)

[1] V. Q. Dang and J. D. Kotulski, "A Distributed-Memory Implementation of Schur-Complement PCA Preconditioner for Ill-Conditioned Problems," *2023 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting (USNC-URSI)*, Portland, OR, USA, 2023, pp. 593-594

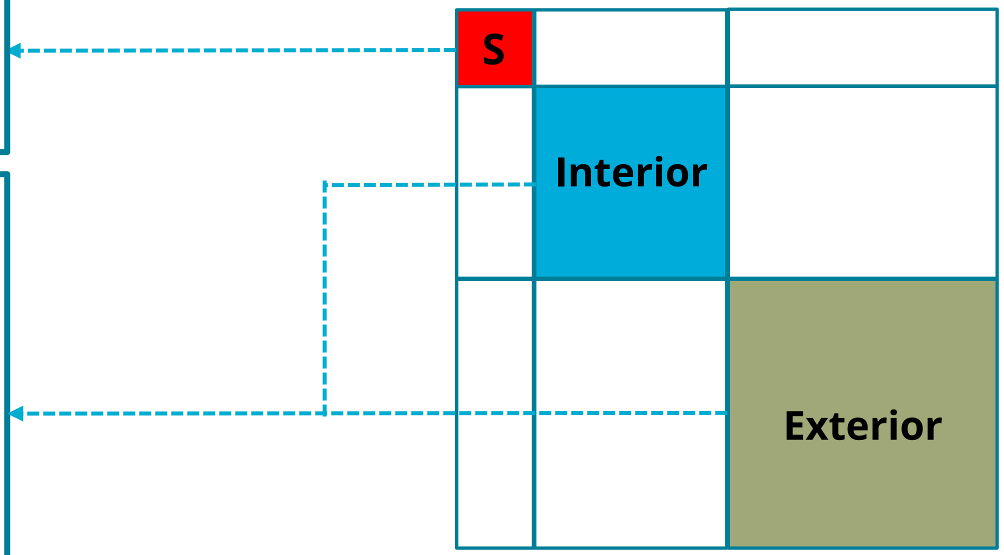
Preconditioners in Multi-Region Solver

- An iterative solution approach (i.e. Krylov subspace method) is used to solve the multi-region linear equation system
  - Generalized Conjugate Residual method (GCR), Generalized Minimum Residual method (GMRES), ...

- Additive Schwarz preconditioner: combines local preconditioners constructed from the different region self-interaction blocks

- Slot region:
  - Slot solver simply performs LU inverse of the matrix block

- Interior/Exterior regions:
  - MoM solver: calculates Schur-complement PCA preconditioner (referred to [1] for details)
  - ACA solver: block-diagonal preconditioner and algebraically-transformed block-diagonal preconditioner
    - Future: multi-resolution preconditioner, sparse approximate inverse (SPAI) preconditioner



[1] V. Q. Dang and J. D. Kotulski, "A Distributed-Memory Implementation of Schur-Complement PCA Preconditioner for Ill-Conditioned Problems," *2023 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting (USNC-URSI)*, Portland, OR, USA, 2023, pp. 593-594

Algebraically-Transformed Block-Diagonal Preconditioner

- Inspired by the simple algebraic method [1] for solving MoM matrices iteratively
  - [1] seeks solution of unknowns $X \approx Z^{-1} Y$
  - Cannot be applied to cavity problems as a whole because there are two close basis functions located in two different regions, i.e. no interaction
- Our approach: used as a preconditioner to find results of $M^{-1}(Y - ZX)$ in the ACA solver
- Procedure:
  - Renumber the basis functions using a distance criterion measured from a reference point
  - Use oct-tree or binary-tree to partition and group basis functions to boxes
  - Transform $\boldsymbol{ZX} = \boldsymbol{Y}$ to $\overline{\boldsymbol{Z}}\boldsymbol{X} = \overline{\boldsymbol{Y}}$ by zeroing the two blocks closest to diagonal blocks
  - Find $\boldsymbol{X} = \overline{\boldsymbol{Z}}^{-1}\overline{\boldsymbol{Y}}$

[1] Sadasiva M. Rao and Charles F. Nelatury, " A Simple Algebraic Method to Solve the Matrix Equation Generated by the Method of Moments," ACES, Orlando, Florida, 2024

Algebraic Transformation (1/2)

- Z-matrix can be written as:

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{Z}_{11} & \boldsymbol{Z}_{12} & \boldsymbol{Z}_{13} & \boldsymbol{Z}_{14} & \cdots & \boldsymbol{Z}_{1P} \\ \boldsymbol{Z}_{21} & \boldsymbol{Z}_{22} & \boldsymbol{Z}_{23} & \boldsymbol{Z}_{24} & \cdots & \boldsymbol{Z}_{2P} \\ \boldsymbol{Z}_{31} & \boldsymbol{Z}_{32} & \boldsymbol{Z}_{33} & \boldsymbol{Z}_{34} & \cdots & \boldsymbol{Z}_{3P} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{Z}_{P1} & \boldsymbol{Z}_{P2} & \boldsymbol{Z}_{P3} & \boldsymbol{Z}_{P4} & \cdots & \boldsymbol{Z}_{PP} \end{bmatrix}$$

- Transform the first block row to: $\widetilde{\boldsymbol{Z}}\boldsymbol{X} = \widetilde{\boldsymbol{Y}}$ where $\widetilde{\boldsymbol{Z}} = \boldsymbol{R_1Z}, \widetilde{\boldsymbol{Y}} = \boldsymbol{R_1Y}$, and

$$\boldsymbol{R}_1 = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{R}_{12} & \boldsymbol{R}_{13} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{I} \end{bmatrix}$$

Algebraic Transformation (2/2)

- Considering the first row of the $\tilde{\boldsymbol{Z}}$-matrix, we have:

$$\tilde{\boldsymbol{Z}}_{12} = \boldsymbol{Z}_{12} + \boldsymbol{R}_{12}\boldsymbol{Z}_{22} + \boldsymbol{R}_{13}\boldsymbol{Z}_{32}$$
$$\tilde{\boldsymbol{Z}}_{13} = \boldsymbol{Z}_{13} + \boldsymbol{R}_{12}\boldsymbol{Z}_{23} + \boldsymbol{R}_{13}\boldsymbol{Z}_{33}$$

- Next we solve for $\boldsymbol{R}_{12}$ and $\boldsymbol{R}_{13}$ by forcing the elements of $\tilde{\boldsymbol{Z}}_{12}$ and $\tilde{\boldsymbol{Z}}_{13}$ to zero

$$\boldsymbol{Z}_{12} + \boldsymbol{R}_{12}\boldsymbol{Z}_{22} + \boldsymbol{R}_{13}\boldsymbol{Z}_{32} = \boldsymbol{0}$$
$$\boldsymbol{Z}_{13} + \boldsymbol{R}_{12}\boldsymbol{Z}_{23} + \boldsymbol{R}_{13}\boldsymbol{Z}_{33} = \boldsymbol{0}$$

- Applying similar procedures to rows 2, 3, …, P and each time solving a $2M{\times}2M$ matrix to obtain $\overline{\boldsymbol{Z}}\boldsymbol{X} = \overline{\boldsymbol{Y}}$ where the new $\overline{\boldsymbol{Z}}$-matrix is given by

$$\begin{bmatrix} \tilde{\boldsymbol{Z}}_{11} & \boldsymbol{0} & \boldsymbol{0} & \tilde{\boldsymbol{Z}}_{14} & \cdots & \tilde{\boldsymbol{Z}}_{1,P-2} & \tilde{\boldsymbol{Z}}_{1,P-1} & \tilde{\boldsymbol{Z}}_{1P} \\ \boldsymbol{0} & \tilde{\boldsymbol{Z}}_{22} & \boldsymbol{0} & \tilde{\boldsymbol{Z}}_{24} & \cdots & \tilde{\boldsymbol{Z}}_{2,P-2} & \tilde{\boldsymbol{Z}}_{2,P-1} & \tilde{\boldsymbol{Z}}_{2P} \\ \tilde{\boldsymbol{Z}}_{31} & \boldsymbol{0} & \tilde{\boldsymbol{Z}}_{33} & \boldsymbol{0} & \cdots & \tilde{\boldsymbol{Z}}_{3,P-2} & \tilde{\boldsymbol{Z}}_{3,P-1} & \tilde{\boldsymbol{Z}}_{3P} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \tilde{\boldsymbol{Z}}_{P1} & \tilde{\boldsymbol{Z}}_{P2} & \tilde{\boldsymbol{Z}}_{P3} & \tilde{\boldsymbol{Z}}_{P4} & \cdots & \boldsymbol{0} & \boldsymbol{0} & \tilde{\boldsymbol{Z}}_{PP} \end{bmatrix}$$

Solve Transformed Equation

- Re-write the previous equation $\bar{\boldsymbol{Z}}\boldsymbol{X} = \bar{\boldsymbol{Y}}$ as $[\bar{\boldsymbol{Z}}_d + \bar{\boldsymbol{Z}}_f]\boldsymbol{X} = \bar{\boldsymbol{Y}}$ where $\bar{\boldsymbol{Z}}_d$ and $\bar{\boldsymbol{Z}}_f$ represent the diagonal and off-diagonal matrices, respectively

- Step 1: Compute $\boldsymbol{X}_{0,d} = \bar{\boldsymbol{Z}}_d^{-1}\bar{\boldsymbol{Y}} = \bar{\boldsymbol{Z}}_d^{-1}\bar{\boldsymbol{Y}}_{\boldsymbol{0}}$

→ **Algebraically-Transformed Block-Diagonal Preconditioner**

- Step 2: Compute $[\bar{\boldsymbol{Z}}_d + \bar{\boldsymbol{Z}}_f]\boldsymbol{X}_{0,d} = \bar{\boldsymbol{Y}}_{0,d}$

- Step 3: Compute $(\bar{\boldsymbol{Y}}_0 - \bar{\boldsymbol{Y}}_{0,d}) = \bar{\boldsymbol{Y}}_1$

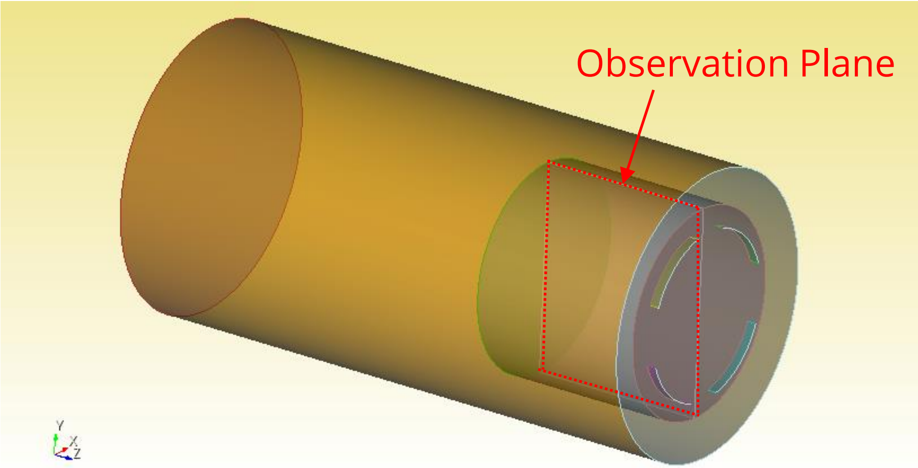
- Repeat these 3 steps *n* times

- Final solution is: $\boldsymbol{X} = \boldsymbol{X}_{0,d} + \boldsymbol{X}_{1,d} + \cdots + \boldsymbol{X}_{n-1,d}$

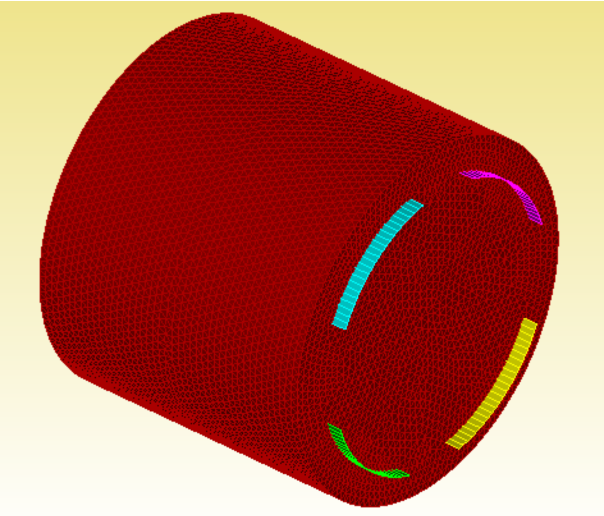
→ **Future Near-field Preconditioner**

Preliminary Results

Simulation Setup and Accuracy Comparison

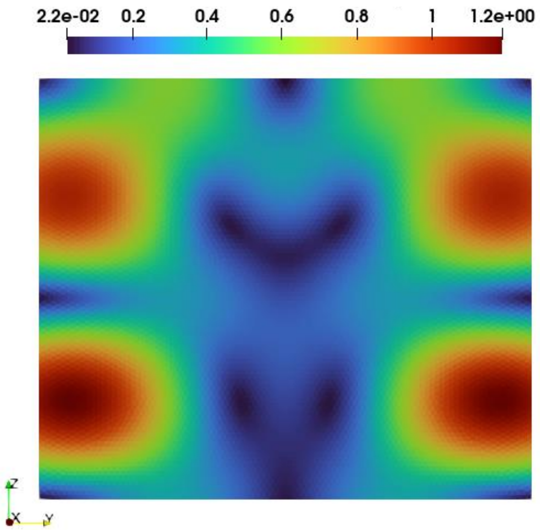


Geometry and Observation Plane

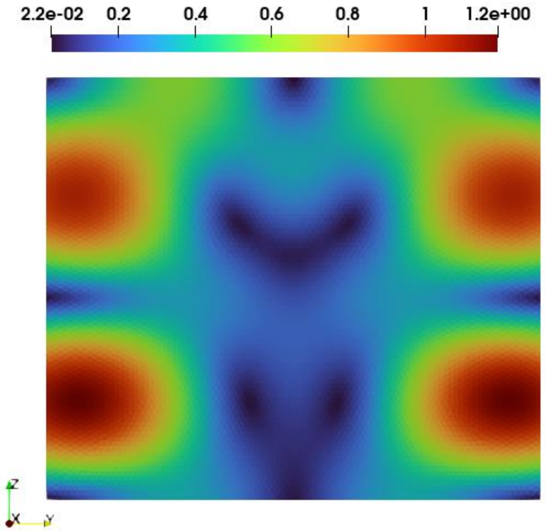


Interior and Slot Meshing

- High-Q factor slotted cylindrical cavity at f = 1. 2GHz (near a resonance) with 126,976 unknowns with three regions
  - Slot region: 88 unknowns
  - Interior region: 29,007 unknowns (leaf box size: 0.44*λ)
  - Exterior region: 97,881 unknowns (leaf box size: 0.6*λ) (~3.3x bigger than the interior)
- Tests run on 2 nodes, 8 ranks per node (1 MI300 GPU per rank, GPU-aware MPI):
  - Near-far distance threshold: 2*box size
  - ACA tolerance: 1e-8
  - GCR solver with tolerance of 1e-10



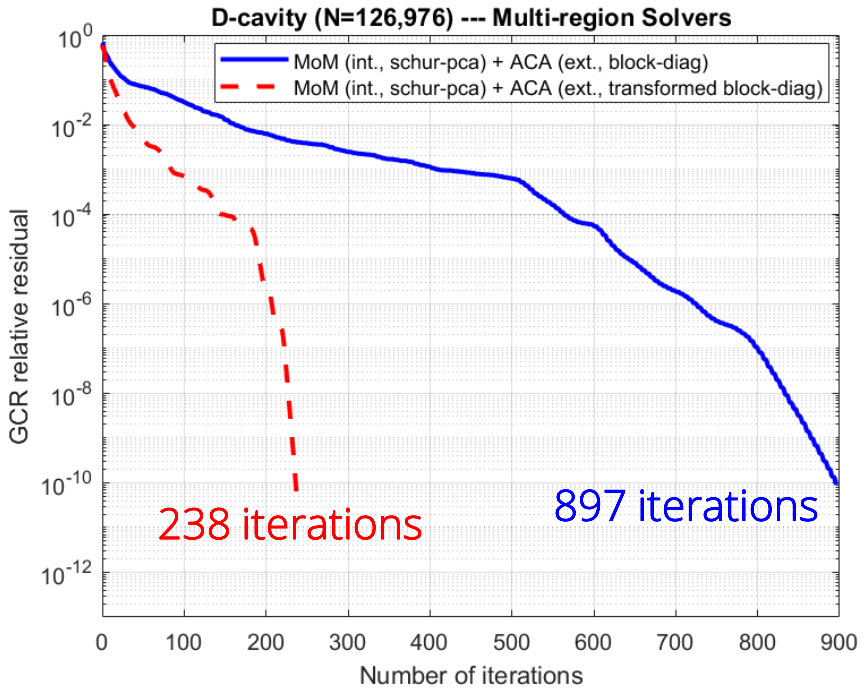
E-field (mag) by multi-region solver with ACA for both interior and exterior



E-field (mag) by direct solver

Preliminary Results

- Comparison:
  1) MoM+Schur-complement PCA preconditioner for the interior and ACA+Block-diagonal preconditioner for the exterior
  2) MoM+Schur-complement PCA preconditioner for the interior and ACA+Transformed-block-diagonal preconditioner for the exterior



Convergence

	(1)	(2)
Interior preconditioner build time (sec.)	41.66	42.48
Exterior preconditioner build time (sec.)	0.79	3.69
GCR solve time (sec.)	90.79	20.20
Interior preconditioner memory (MB)	5362.84	5362.84
Exterior preconditioner memory (MB)	9536.62	27411.23

Time and Memory

Conclusions and Future Work

- Developed a multi-region solver framework for high-Q resonant cavity problems on distributed-memory hardware-accelerated systems with Kokkos+Kokkos Kernels for performance portability
  - Effective in solving ill-conditioned problems
- A new algebraically-transformed block-diagonal preconditioner was developed for the ACA subdomain solver: faster solve time/convergence rate with the increase of preconditioner build time and memory requirement
- Future work:
  - Performance optimization for the algebraically-transformed block-diagonal preconditioner
  - Full evaluation when integration into the main Gemma code completes

Thank You!