THE OHIO STATE UNIVERSITY

SAND2020-6666C

An Approximate Direct Inverse as a Preconditioner For Ill-Conditioned Problems

Chung Hyun Lee, Jin-Fa Lee, The Ohio State University, United States;
William Langston, Brian Zinser, Vinh Dang, Andy Huang, Salvatore Campione, Sandia National Laboratories, United State

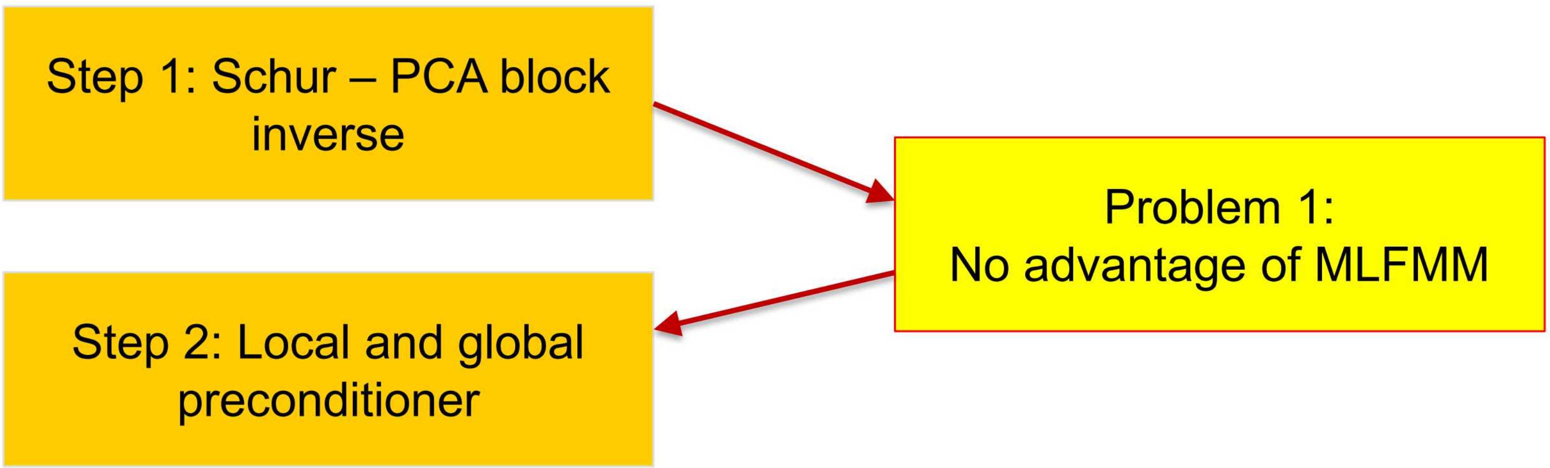
July 6th 2020

Sandia National Laboratories

Acknowledgement: 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

Mission Statement:

Find an effective preconditioner for integral equation



Step 1. Schur – PCA Block Inverse

Mission Statement:

Find an effective preconditioner for integral equation

- Step 1: Schur – PCA block inverse
  - Ingredient 1: Schur Complement
  - Ingredient 2: Principal Component Analysis
  - Ingredient 3: Transpose of MLFMM

Step 1. Schur – PCA Block Inverse

➢ Ingredient 1: Schur-Complement via octree

❖ Matrix Decomposition

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{A}_2 \end{bmatrix}$$

where

- A is system matrix of CFIE
- Let's assume A_1, C_{12}, A_2 and C_{21} are 2 x 2 partition of entire A

$$\mathbf{PAx} = \mathbf{Pb} \longrightarrow \overline{\mathbf{A}}\mathbf{x} = \overline{\mathbf{b}}$$

$$\begin{aligned}\overline{\mathbf{A}} &= \begin{bmatrix} \mathbf{A}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{A}_1 & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{A}_2 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I} & \mathbf{A}_1^{-1}\mathbf{C}_{12} \\ \mathbf{A}_2^{-1}\mathbf{C}_{21} & \mathbf{I} \end{bmatrix} \xrightarrow{\text{Schur Decomposition}} \begin{bmatrix} \mathbf{I} & \mathbf{A}_1^{-1}\mathbf{C}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} - \mathbf{A}_1^{-1}\mathbf{C}_{12}\mathbf{A}_2^{-1}\mathbf{C}_{21} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{A}_2^{-1}\mathbf{C}_{21} & \mathbf{I} \end{bmatrix}\end{aligned}$$

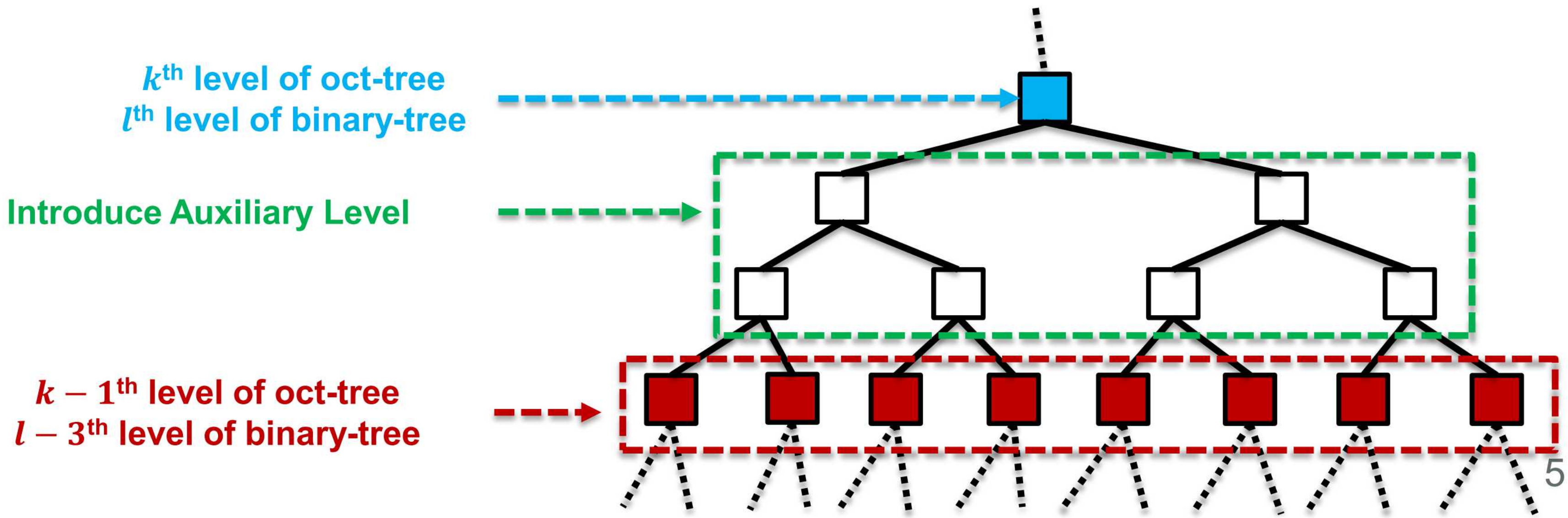
Step 1. Schur – PCA Block Inverse

➢ Ingredient 1: Schur-Complement via octree

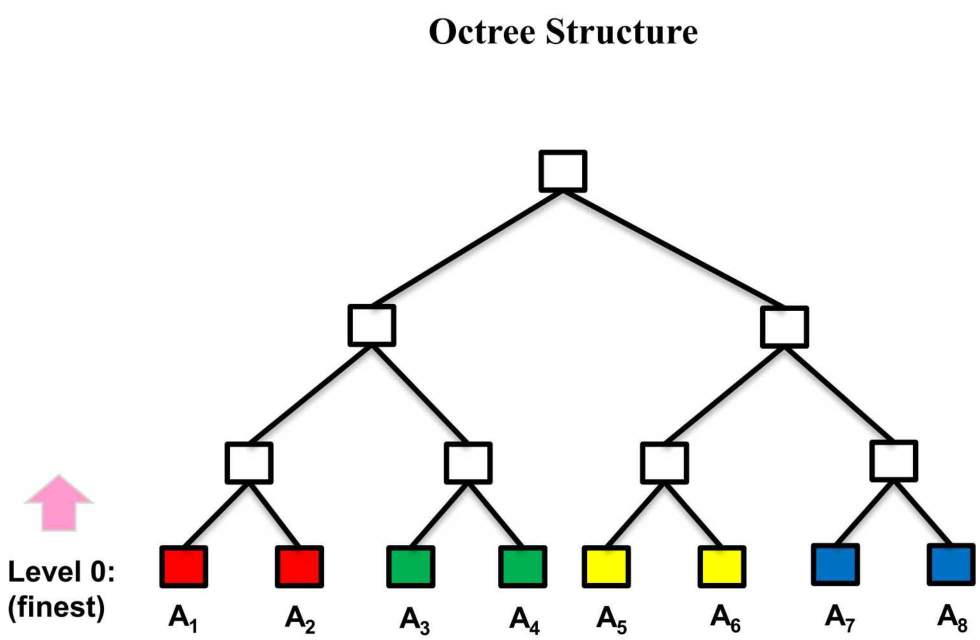
$$Let \quad A_i^l = \begin{bmatrix} A_j^{l-1} & C_{jk}^{l-1} \\ C_{kj}^{l-1} & A_k^{l-1} \end{bmatrix}$$

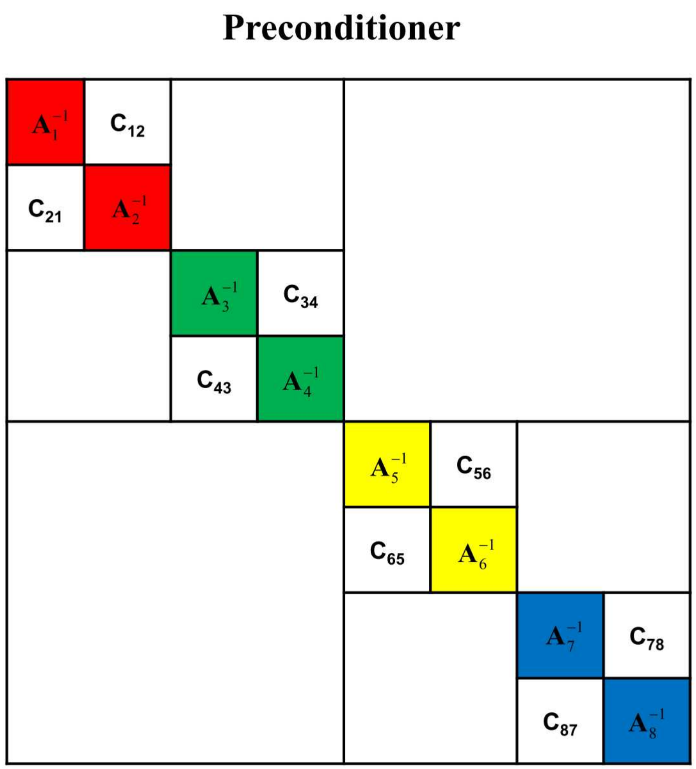
$$\text{Then} \quad (A_i^l)^{-1} = \begin{bmatrix} I_m & 0 \\ A_k^{-1}C_{kj} & I_n \end{bmatrix} \begin{bmatrix} (I_m - A_j^{-1}C_{jk}A_k^{-1}C_{kj})^{-1} & 0 \\ 0 & I_n \end{bmatrix} \begin{bmatrix} I_m & A_j^{-1}C_{jk} \\ 0 & I_n \end{bmatrix} \begin{bmatrix} A_j^{-1} & 0 \\ 0 & A_k^{-1} \end{bmatrix}$$

Note: A_j^{-1} and A_k^{-1} are recursive call until reach to the leaf level



Pictorial explanation for recursive factorization: The finest level

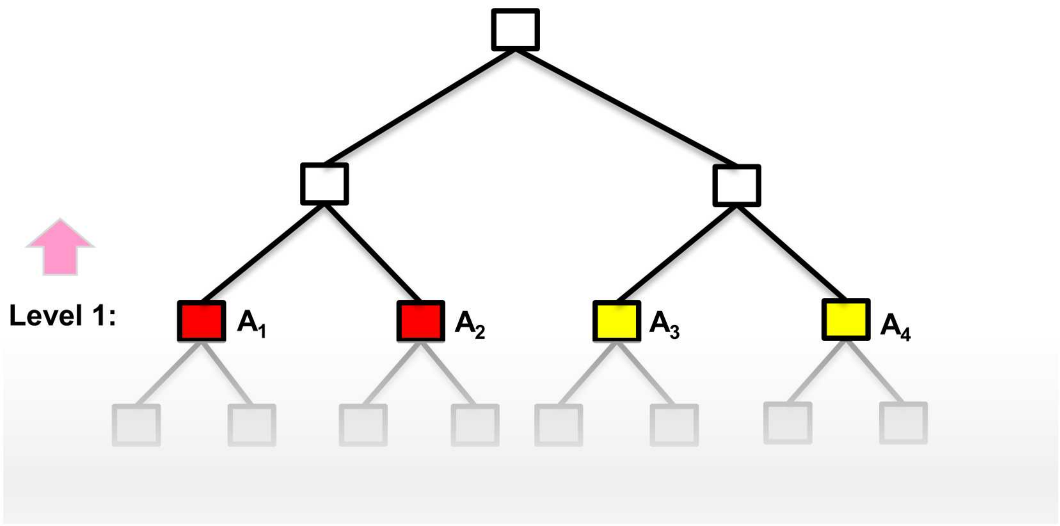




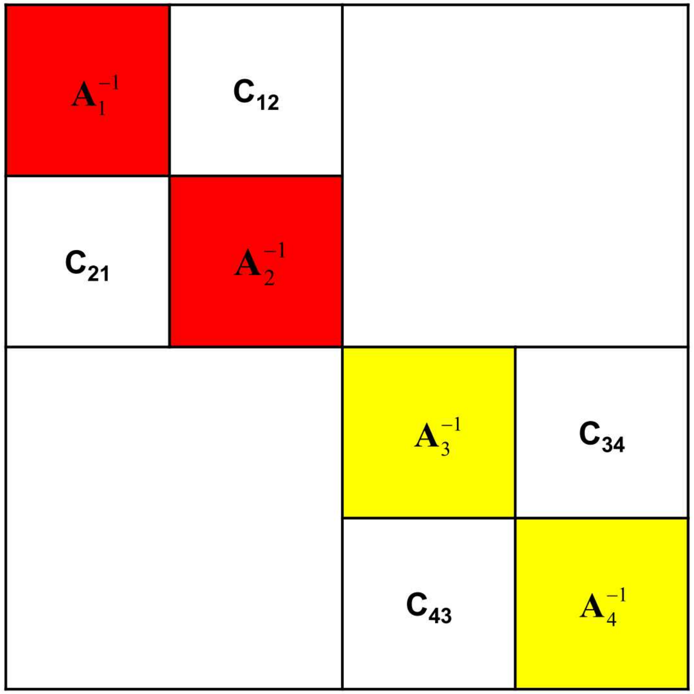
Schur PC with Octree

Pictorial explanation for recursive factorization: Intermediate level

Octree Structure

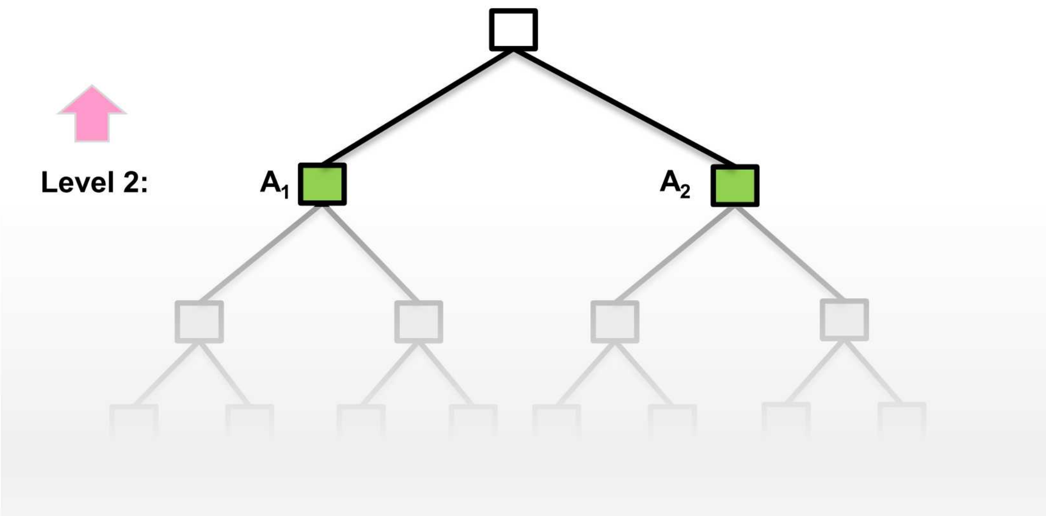


Preconditioner



Pictorial explanation for recursive factorization: Intermediate level

Octree Structure

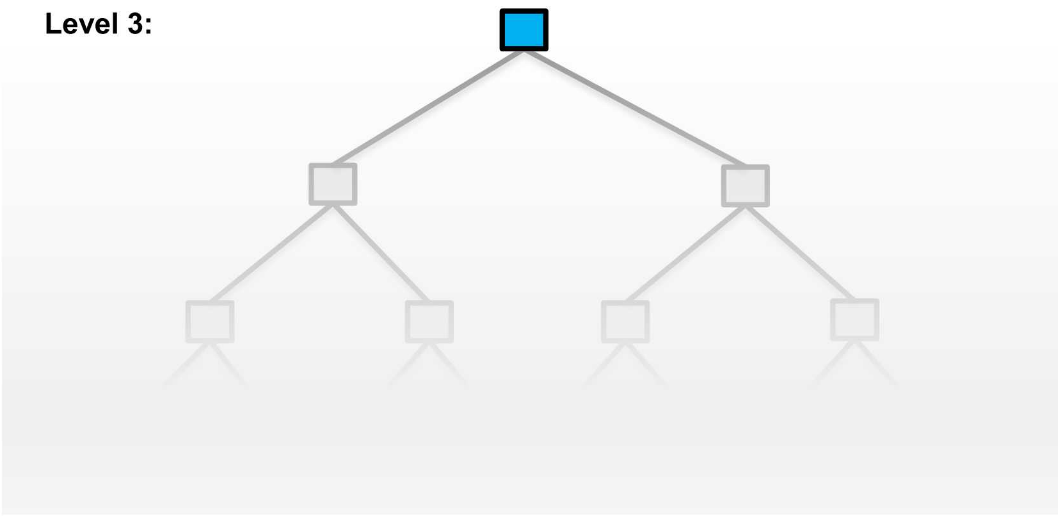


Preconditioner

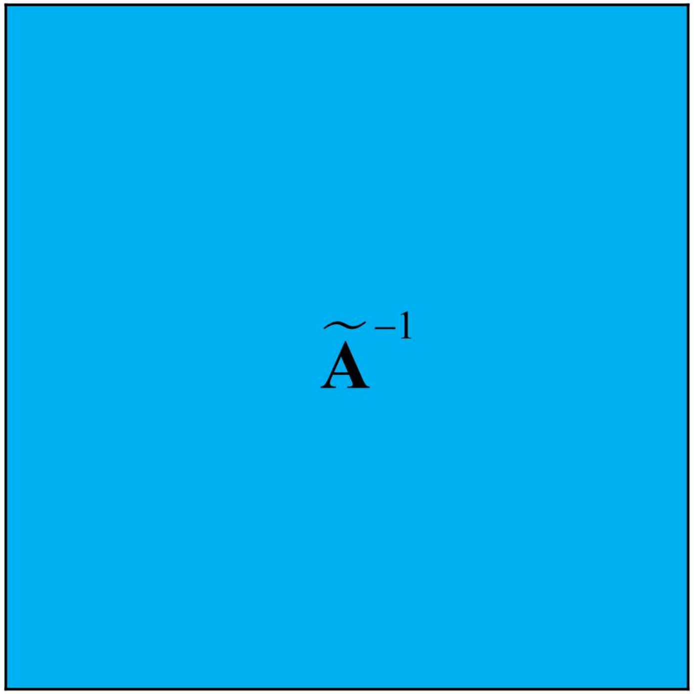
$$\begin{bmatrix} A_1^{-1} & C_{12} \\ C_{21} & A_2^{-1} \end{bmatrix}$$

Pictorial explanation for recursive factorization: The Coarsest level

Octree Structure



Preconditioner



Step 1. Schur – PCA Block Inverse

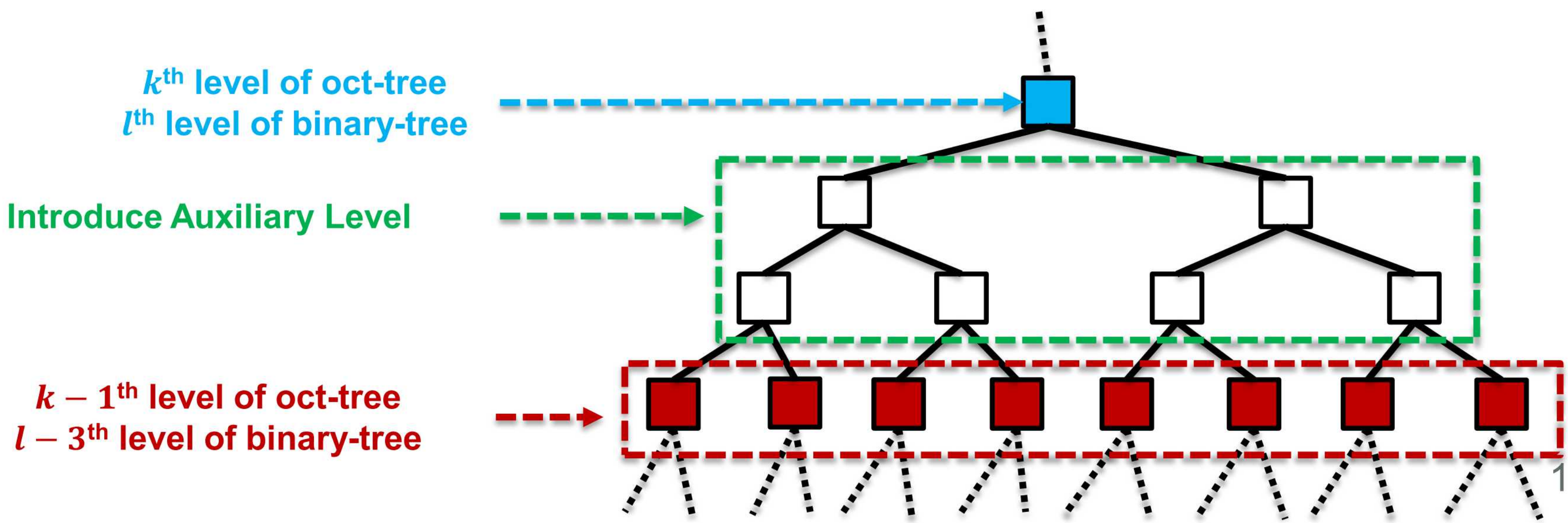
➢ Ingredient 2: Compress with PCA

$$Let \quad A_i^l = \begin{bmatrix} A_j^{l-1} & C_{jk}^{l-1} \\ C_{kj}^{l-1} & A_k^{l-1} \end{bmatrix}$$

Compress Here with
PCA + matrix identity

$$\text{Then} \quad (A_i^l)^{-1} = \begin{bmatrix} I_m & 0 \\ A_k^{-1}C_{kj} & I_n \end{bmatrix} \begin{bmatrix} \boxed{(I_m - A_j^{-1}C_{jk}A_k^{-1}C_{kj})^{-1}} & 0 \\ 0 & I_n \end{bmatrix} \begin{bmatrix} I_m & A_j^{-1}C_{jk} \\ 0 & I_n \end{bmatrix} \begin{bmatrix} A_j^{-1} & 0 \\ 0 & A_k^{-1} \end{bmatrix}$$

Note: A_j^{-1} and A_k^{-1} are recursive call until reach to the leaf level



Step 1. Schur – PCA Block Inverse

➢ Ingredient 2: Compress with PCA

Woodbury Matrix Identity

$$(I_m - \underline{A_j^{-1}C_{jk}A_k^{-1}C_{kj}})^{-1} \simeq (I - \underline{U\Sigma V^H})^{-1} = I + U(\Sigma^{-1} - V^H U)^{-1}V^H = I + USV^H$$

Compress with PCA (Low-rank)

$$S := (\Sigma^{-1} - V^H U)^{-1}$$

U : $m \times q$ matrix with left singular vectors

Σ : q $\times\, q$ diagonal matrix with singular values

V : $q \times n$ matrix with right singular vectors

S : $q \times q$ matrix → Compute explicitly due to $q \ll m, n$

How to determine q? → $\sigma_q / \sigma_1 \leq \epsilon$ where ϵ is predetermined tolerance

Step 1. Schur – PCA Block Inverse

➢ Ingredient 3: Transpose of MLFMM

During the PCA process, one need to compute a matrix-vector-multiplication and can accelerate with MLFMM

$$A_j^{-1} C_{jk} A_k^{-1} C_{kj} \longrightarrow USV^H$$

Need $\mathbf{C}_{jk} x_j$, $\mathbf{C}_{kj} x_k$, $\mathbf{C}_{jk}^T x_k$ and $\mathbf{C}_{kj}^T x_j$ with MLFMM MVM
(No explicitly form a coupling matrix)

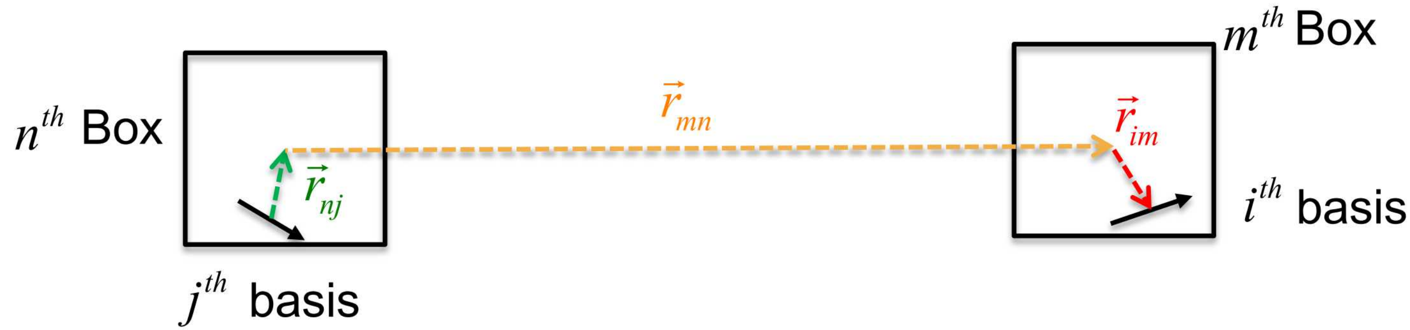
How to compute?

Change

1. **direction of displacement vectors and propagation vectors**
2. **Order of radiation pattern and receiving pattern**

Ingredient 3: Transpose of MLFMM

Example of EFIE and MFIE term with single level FMM,



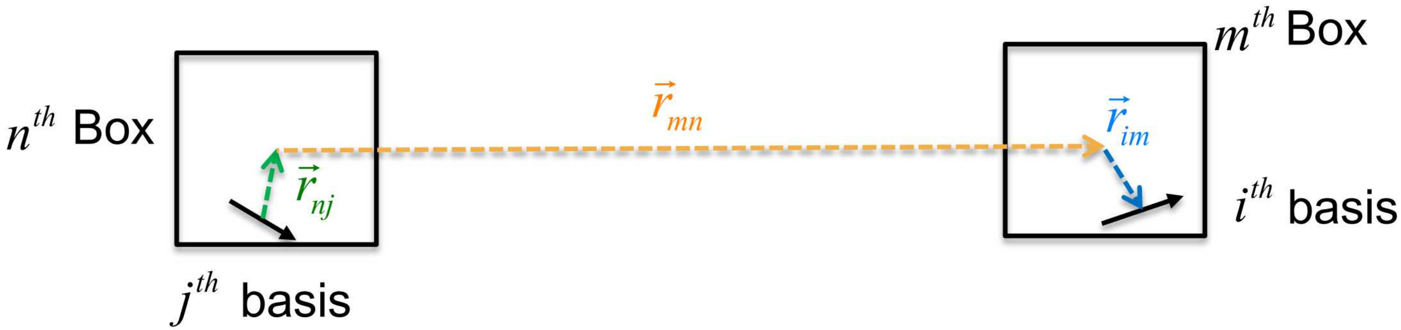
Regular FMM

$$C_{ij}^{EFIE} = \int V_f^E(\hat{k}, \vec{r}_{im}) T_{mn}(\hat{k} \cdot \vec{r}_{mn}) V_s^E(\hat{k}, \vec{r}_{nj}) d\hat{k}^2$$

where V_s^E, T_{mn}, V_f^E, and V_f^M are functions to compute radiation pattern, translation, and receiving pattern for EFIE and MFIE

Ingredient 3: Transpose of MLFMM

Example of EFIE and MFIE term with single level FMM,



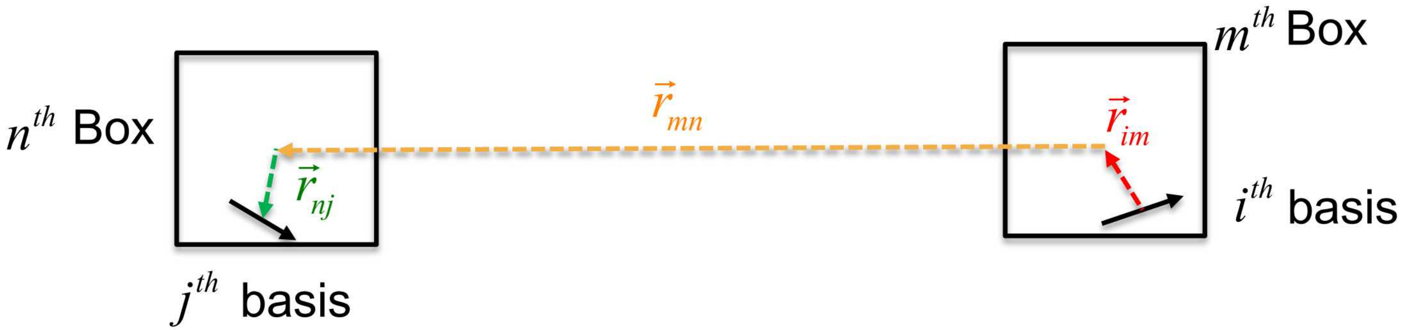
Regular FMM

$$\begin{cases} C_{ij}^{EFIE} = \int V_f^E(\hat{k}, \vec{r}_{im}) T_{mn}(\hat{k} \cdot \vec{r}_{mn}) V_s^E(\hat{k}, \vec{r}_{nj}) d\hat{k}^2 \\ C_{ij}^{MFIE} = \int V_f^M(\hat{k}, \vec{r}_{im}) T_{mn}(\hat{k} \cdot \vec{r}_{mn}) V_s^E(\hat{k}, \vec{r}_{nj}) d\hat{k}^2 \end{cases}$$

where $V_s^E, T_{mn}, V_f^E,$ and V_f^M are functions to compute radiation pattern, translation, and receiving pattern for EFIE and MFIE

➢ Ingredient 3: Transpose of MLFMM

Example of EFIE and MFIE term with single level FMM,



Regular FMM

$$\begin{cases} C_{ij}^{EFIE} = \int V_f^E(\hat{k}, \vec{r}_{im}) T_{mn}(\hat{k} \cdot \vec{r}_{mn}) V_s^E(\hat{k}, \vec{r}_{nj}) d\hat{k}^2 \\ C_{ij}^{MFIE} = \int V_f^M(\hat{k}, \vec{r}_{im}) T_{mn}(\hat{k} \cdot \vec{r}_{mn}) V_s^E(\hat{k}, \vec{r}_{nj}) d\hat{k}^2 \end{cases}$$

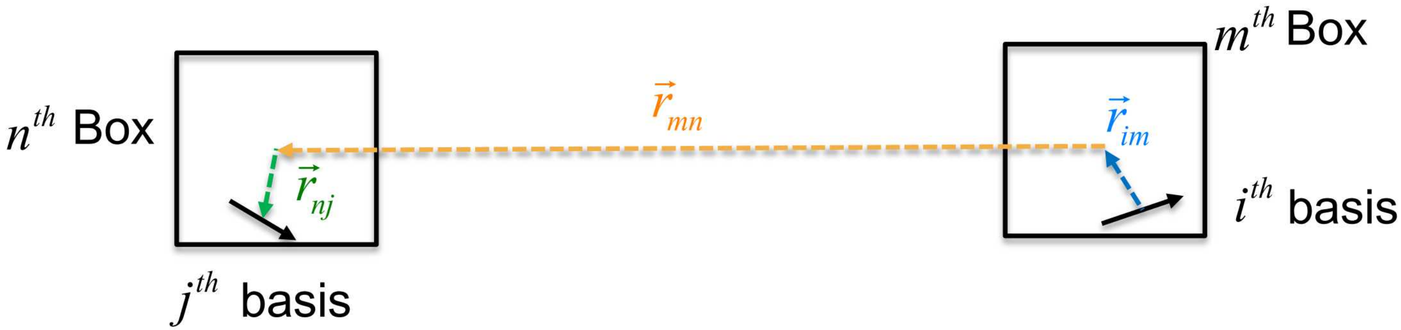
Transpose of FMM

$$\begin{cases} (C_{ij}^{EFIE})^T = \int V_s^E(-\hat{k}, -\vec{r}_{nj}) T_{mn}(-\hat{k} \cdot -\vec{r}_{mn}) V_f^E(-\hat{k}, -\vec{r}_{im}) d\hat{k}^2 \\ \end{cases}$$

where $V_s^E, T_{mn}, V_f^E,$ and V_f^M are functions to compute radiation pattern, translation, and receiving pattern for EFIE and MFIE

Ingredient 3: Transpose of MLFMM

Example of EFIE and MFIE term with single level FMM,



Regular FMM

$$\begin{cases} C_{ij}^{EFIE} = \int V_f^E(\hat{k}, \vec{r}_{im}) T_{mn}(\hat{k} \cdot \vec{r}_{mn}) V_s^E(\hat{k}, \vec{r}_{nj}) d\hat{k}^2 \\ C_{ij}^{MFIE} = \int V_f^M(\hat{k}, \vec{r}_{im}) T_{mn}(\hat{k} \cdot \vec{r}_{mn}) V_s^E(\hat{k}, \vec{r}_{nj}) d\hat{k}^2 \end{cases}$$

Transpose of FMM

$$\begin{cases} (C_{ij}^{EFIE})^T = \int V_s^E(-\hat{k}, -\vec{r}_{nj}) T_{mn}(-\hat{k} \cdot -\vec{r}_{mn}) V_f^E(-\hat{k}, -\vec{r}_{im}) d\hat{k}^2 \\ (C_{ij}^{MFIE})^T = \int V_s^E(-\hat{k}, -\vec{r}_{nj}) T_{mn}(-\hat{k} \cdot -\vec{r}_{mn}) V_f^M(-\hat{k}, -\vec{r}_{im}) d\hat{k}^2 \end{cases}$$

where $V_s^E, T_{mn}, V_f^E,$ and V_f^M are functions to compute radiation pattern, translation, and receiving pattern for EFIE and MFIE

Step 1. Schur – PCA Block Inverse

➢ Ingredient 3: Transpose of MLFMM

- Radiation and Receiving pattern of EFIE

$$V_s^E(\hat{k},\vec{r}_{nj}) = \int (\bar{\bar{I}} - \boxed{\hat{k}\hat{k})} \cdot \alpha_j(r) e^{-jk\boxed{\hat{k}\cdot\vec{r}_{nj}}} dS = V_s^M(-\hat{k},-\vec{r}_{nj})$$

$$V_f^E(\hat{k},\vec{r}_{im}) = \int (\bar{\bar{I}} - \boxed{\hat{k}\hat{k})} \cdot \alpha_i(r) e^{-jk\boxed{\hat{k}\cdot\vec{r}_{im}}} dS = V_f^E(-\hat{k},-\vec{r}_{im})$$

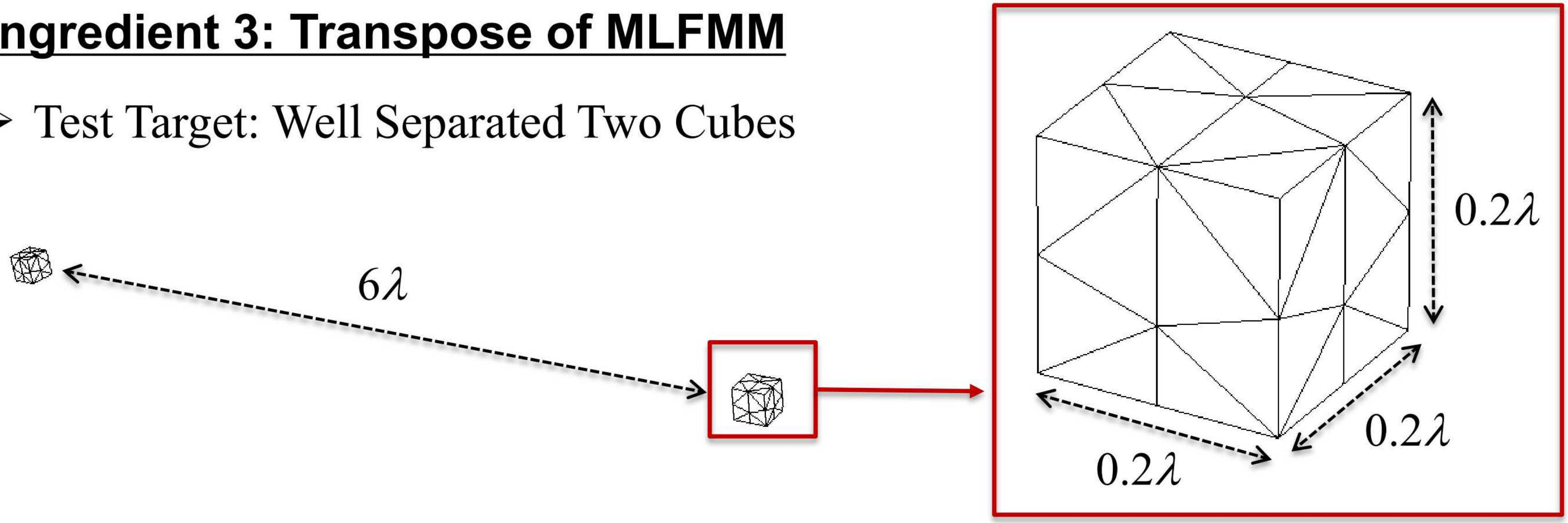
- Receiving pattern of MFIE

$$V_f^M(\hat{k},\vec{r}_{im}) = -\boxed{\hat{k}} \times \int e^{-jk\boxed{\hat{k}\cdot\vec{r}_{im}}} (\alpha_i(r) \times \hat{n}) dS$$

$$V_f^M(-\hat{k},-\vec{r}_{im}) = \boxed{\hat{k}} \times \int e^{-jk\boxed{\hat{k}\cdot\vec{r}_{im}}} (\alpha_i(r) \times \hat{n}) dS = -V_f^M(\hat{k},\vec{r}_{im})$$

Step 1. Schur – PCA Block Inverse

- **Ingredient 3: Transpose of MLFMM**
  - Test Target: Well Separated Two Cubes



- Relative Error:

$$\delta_{Zx} := \frac{\left\| Z_{CFIE}\bar{x} - Z_{MLFMM}\bar{x} \right\|}{\left\| Z_{CFIE}\bar{x} \right\|} = 2.488 \times 10^{-2}$$

$$\delta_{ZTx} := \frac{\left\| Z^{T}{}_{CFIE}\bar{x} - Z^{T}{}_{MLFMM}\bar{x} \right\|}{\left\| Z^{T}{}_{CFIE}\bar{x} \right\|} = 3.054 \times 10^{-2}$$

* Assembled Explicitly

$$Z_{CFIE} = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} \quad Z_{FMM} = \begin{bmatrix} Z_{11} & Z_{12}{}^{FMM} \\ Z_{21}{}^{FMM} & Z_{22} \end{bmatrix}$$

Z_{11}, Z_{22} : Near Coupling (Zero out)

Z_{12}, Z_{21} : Far Coupling

$Z_{12}{}^{FMM}, Z_{21}{}^{FMM}$: MLFMM Coupling

Step 1. Schur – PCA Block Inverse

- Matrix Vector Multiplication

$$\mathbf{A} = \begin{bmatrix} \mathbf{A_1} & \mathbf{C_{12}} \\ \mathbf{C_{21}} & \mathbf{A_2} \end{bmatrix}$$

Let $\mathbf{S} = \left(\mathbf{D}^{-1} - \mathbf{RL}\right)^{-1}$

$$[y] = [\mathbf{A}]^{-1}[x] = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{A}_2^{-1}\mathbf{C}_{21} & \mathbf{I} \end{bmatrix} \begin{bmatrix} S & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{A}_1^{-1}\mathbf{C}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{A}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2^{-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},$$

$$= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{A}_2^{-1}\mathbf{C}_{21} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{A}_1^{-1}\mathbf{C}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{A}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2^{-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$= \begin{bmatrix} \underbrace{\boxed{\mathbf{S}\left[\overset{\textcircled{2}}{\boxed{\mathbf{A}_1^{-1}x_1}} - \mathbf{A}_1^{-1}\mathbf{C}_{12}\overset{\textcircled{1}}{\boxed{\mathbf{A}_2^{-1}x_2}}\right]}}_{\textcircled{3}} \\ \underbrace{-\mathbf{A}_2^{-1}\mathbf{C}_{21}\overset{\textcircled{3}}{\boxed{\mathbf{S}\left[\mathbf{A}_1^{-1}x_1 - \mathbf{A}_1^{-1}\mathbf{C}_{12}\mathbf{A}_2^{-1}x_2\right]}} + \overset{\textcircled{1}}{\boxed{\mathbf{A}_2^{-1}x_2}}}_{\textcircled{4}} \end{bmatrix}$$

① Recursive MVM call for $\mathbf{A}_2^{-1}x_2$

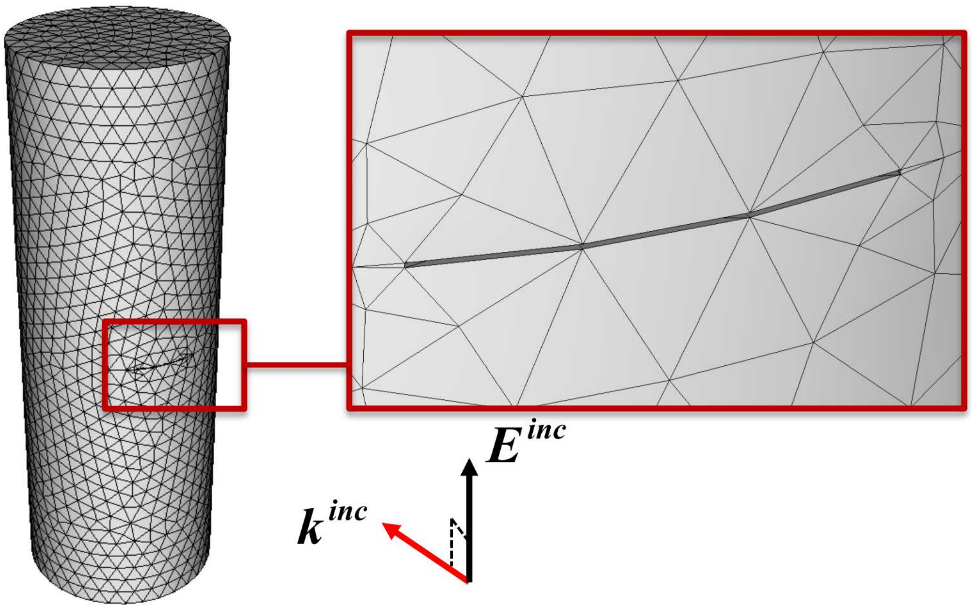
② Recursive MVM call for $\mathbf{A}_1^{-1}x_1$

③ Update $\mathbf{S}\left[\mathbf{A}_1^{-1}x_1 - \mathbf{A}_1^{-1}\mathbf{C}_{12}\mathbf{A}_2^{-1}x_2\right]$ with ① and ②

④ Update $-\mathbf{A}_2^{-1}\mathbf{C}_{21}\mathbf{S}\left[\mathbf{A}_1^{-1}x_1 - \mathbf{A}_1^{-1}\mathbf{C}_{12}\mathbf{A}_2^{-1}x_2\right] + \mathbf{A}_2^{-1}x_2$ with ① and ③

Step 1. Schur – PCA Block Inverse

- Example: PEC Slotted Cylinder



- Cylinder height h=0.6096m, inner radius=0.1016m
- Metal thickness 0.00635m
- Aperture slot 0.000508m
- Analyze SE around first TM resonance at 1.1294GHz

Inner wall modes

```
TMz010: 1.129391 [GHz]
TMz011: 1.155849 [GHz]
TMz012: 1.231818 [GHz]
TMz013: 1.34896 [GHz]
TMz014: 1.497643 [GHz]

TEz111: 0.8988588 [GHz]
TEz112: 0.9946542 [GHz]
TEz113: 1.136509 [GHz]
TEz114: 1.30954 [GHz]
```

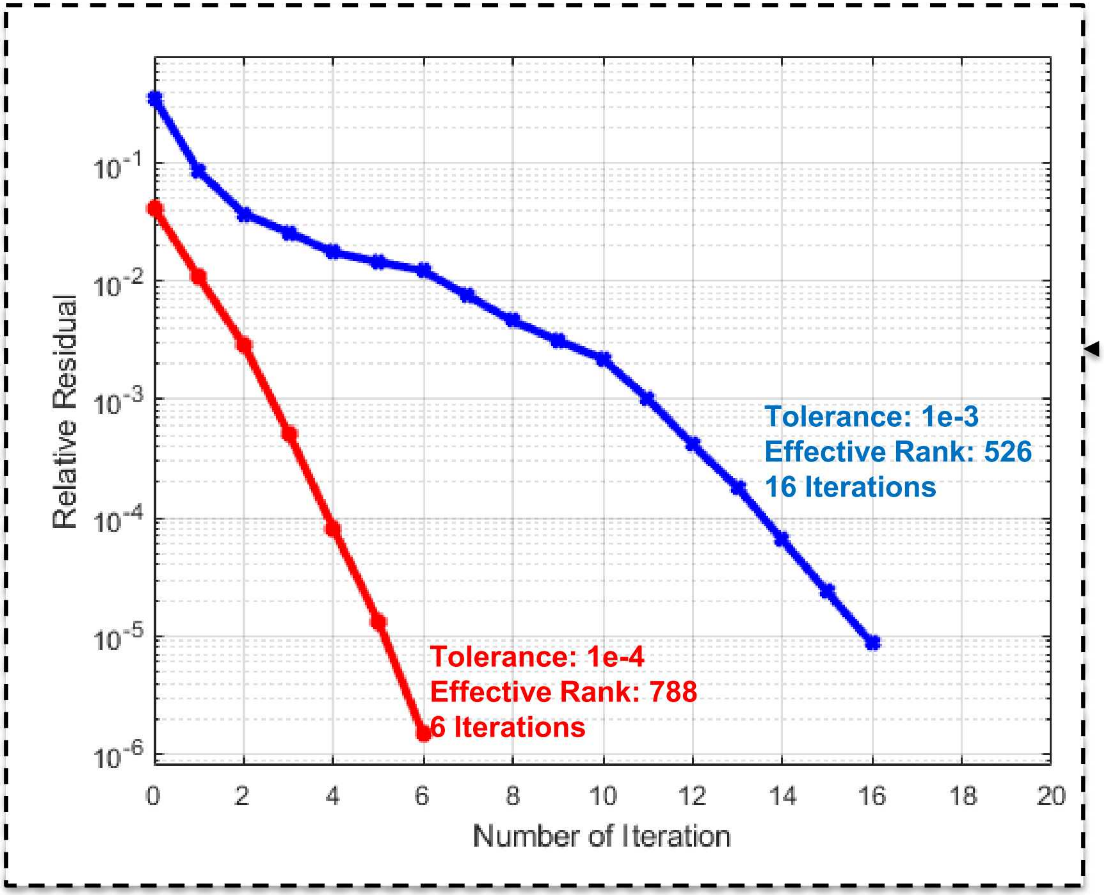
Outer wall modes

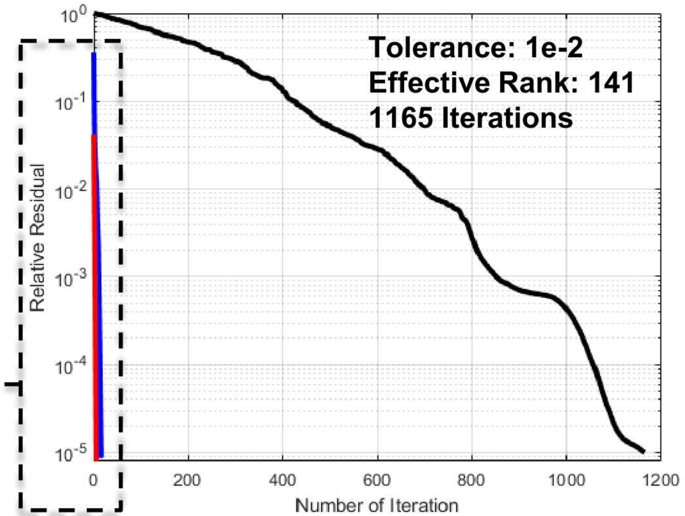
```
TMz010: 1.062956 [GHz]
TMz011: 1.091026 [GHz]
TMz012: 1.171208 [GHz]
TMz013: 1.293849 [GHz]
TMz014: 1.448201 [GHz]

TEz111: 0.8500555 [GHz]
TEz112: 0.9507808 [GHz]
TEz113: 1.098317 [GHz]
TEz114: 1.276536 [GHz]
```

Step 1. Schur – PCA Block Inverse

- Example: PEC Slotted Cylinder (Different PCA tolerance)





- PEC
- Freq: 1132.43 MHz
- # of Basis: 28,944 (~ $\lambda/15$ Mesh)
- The finest block size: 0.3
- The number of oct-tree level: 4
- EFIE + IEDG (with MLFMM)
- Iterative Solver (GCR, $\varepsilon_k = 10^{-5}$)

Step 2. Two Level Preconditioner

Mission Statement:

Find an effective preconditioner for integral equation

Propose two level preconditioner

For given matrix equation, $Ax = b$

We define two preconditioner Q_k and P_l and solve for x the following equation:

$$Q_k P_l A x = Q_k P_l \mathrm{b}$$

where

P_l: Local Preconditioner

Q_k: Global Preconditioner

Step 2. Two Level Preconditioner

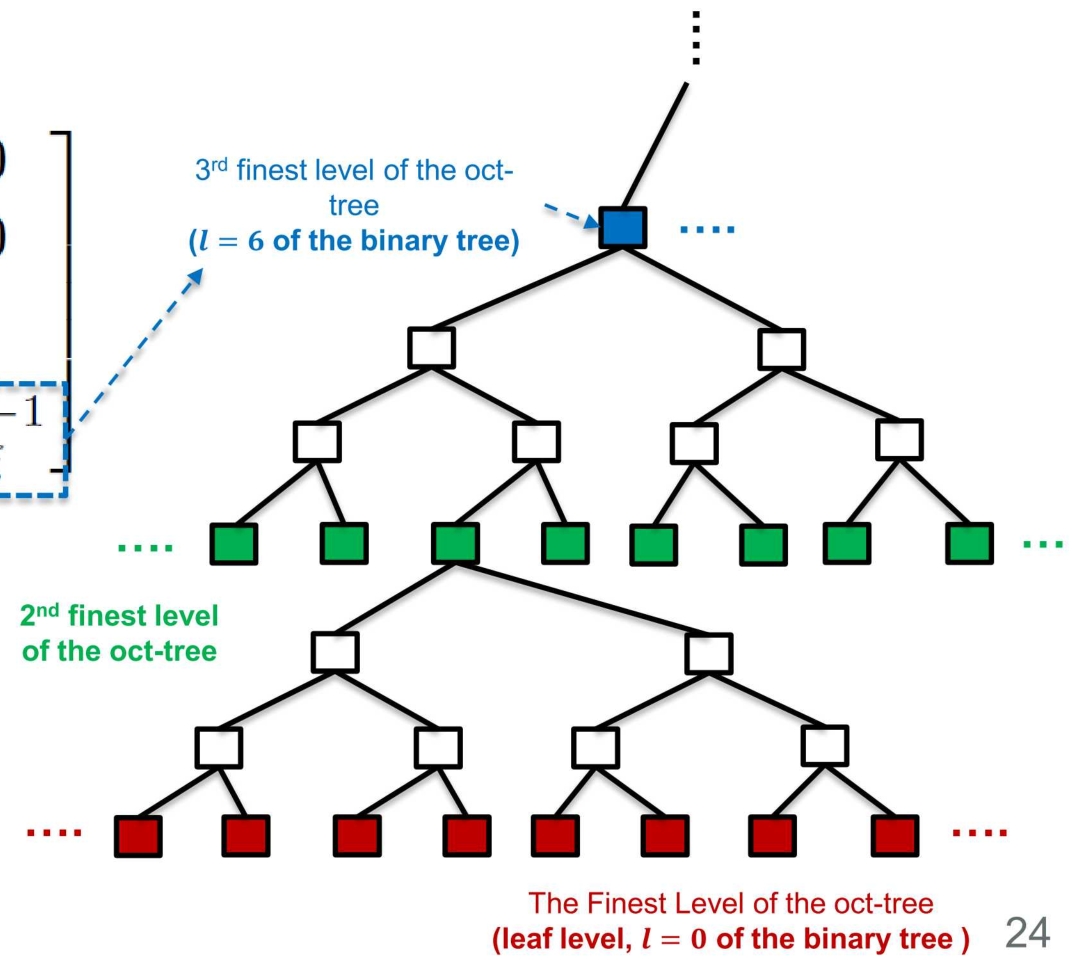
➢ <u>Ingredient 1: Local Preconditioner</u>

- Schur – PCA Block Inverse at the l level of the binary-tree

For example:

$$P_l := \begin{bmatrix} A_1^{-1} & 0 & \cdots & 0 \\ 0 & A_2^{-1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boxed{A_\xi^{-1}} \end{bmatrix}$$

ξ is the number of boxes at the level $l = 6$.



Step 2. Two Level Preconditioner

➢ <u>Ingredient 2: Global Preconditioner</u>

- Compress off-diagonal terms with PCA

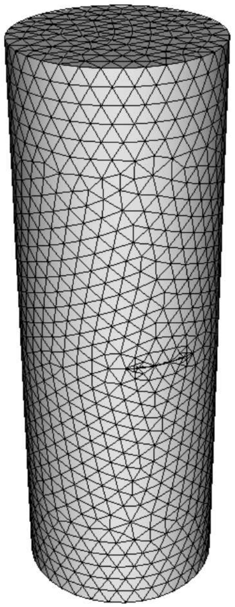
$$Q_k \coloneqq (P_l A)^{-1} = (I+\delta)^{-1} \cong \big(I + \underbrace{L^{m\times k} D^{k\times k} R^{k\times m}}_{\text{Low-rank Approximation}}\big)^{-1} = I - L(D^{-1} + RL)^{-1} R$$

- P_l is the local preconditioner
- δ is off-diagonal terms of matrix A
- $(D^{-1} + RL)^{-1}$ will compute explicitly ($\mathrm{k} \ll m$)
- k fixed to 50 to minimize computation time

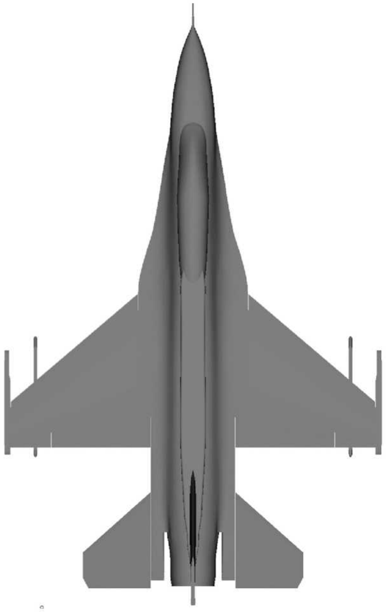
Step 2. Two Level Preconditioner

Applications of Step 2
(two level preconditioner)

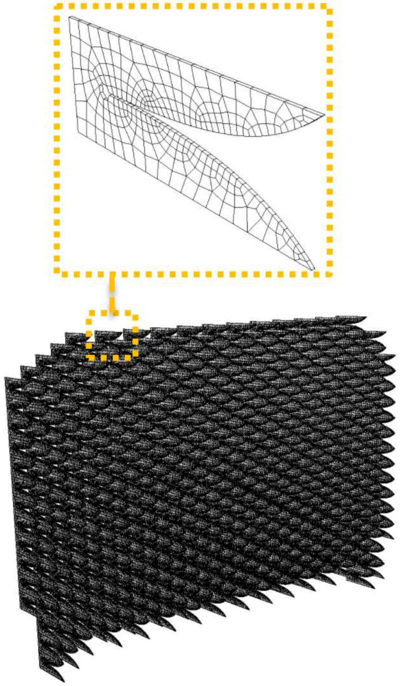
Example 1: IBC
Slotted Cylinder
(at resonance freq)



Example 2: PEC F-16

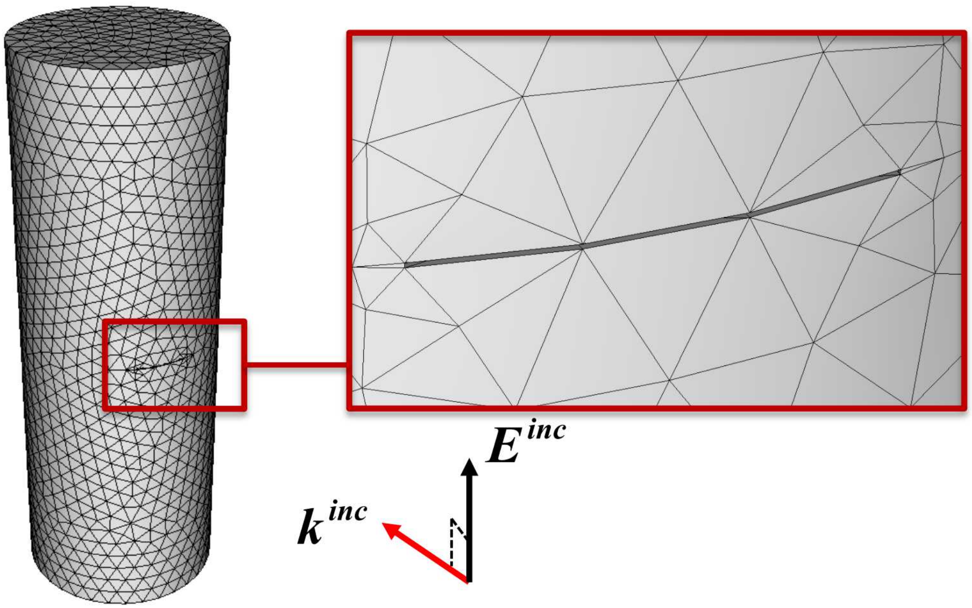


Example 3: PEC
Vivaldi Antenna



Step 2. Two Level Preconditioner

- Example: IBC Slotted Cylinder



- Cylinder height h=0.6096m, inner radius=0.1016m
- Metal thickness 0.00635m
- Aperture slot 0.000508m
- Analyze SE around first TM resonance at 1.1294GHz

Inner wall modes

```
TMz010: 1.129391 [GHz]
TMz011: 1.155849 [GHz]
TMz012: 1.231818 [GHz]
TMz013: 1.34896 [GHz]
TMz014: 1.497643 [GHz]

TEz111: 0.8988588 [GHz]
TEz112: 0.9946542 [GHz]
TEz113: 1.136509 [GHz]
TEz114: 1.30954 [GHz]
```

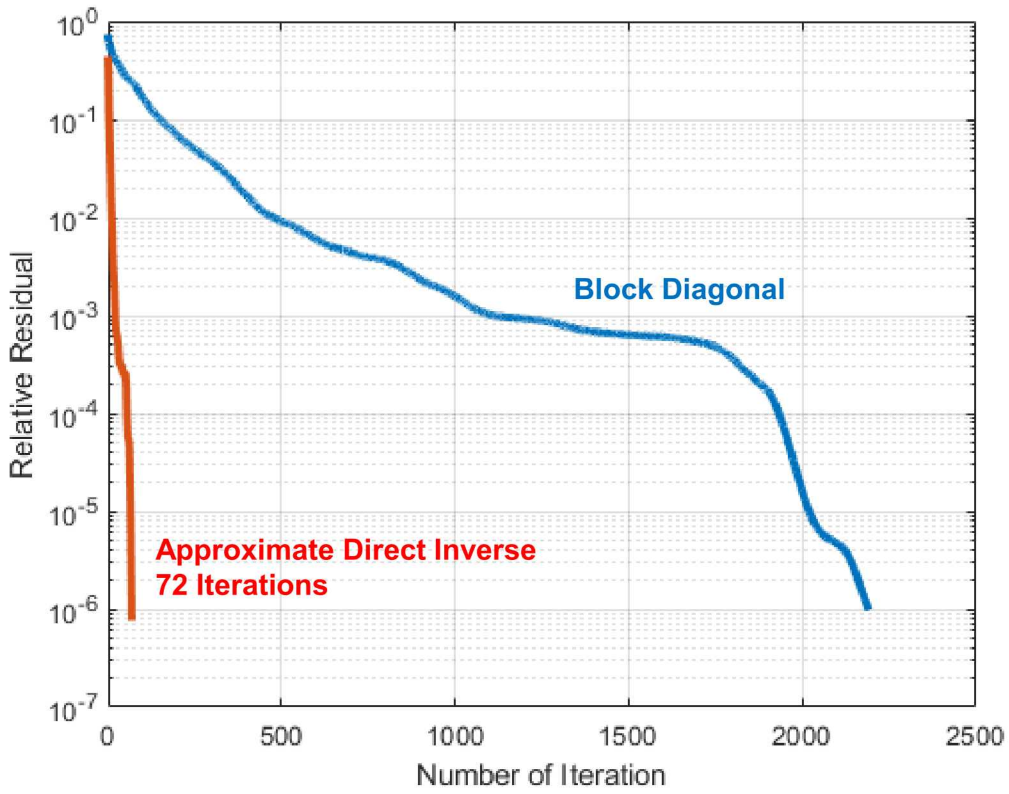
Outer wall modes

```
TMz010: 1.062956 [GHz]
TMz011: 1.091026 [GHz]
TMz012: 1.171208 [GHz]
TMz013: 1.293849 [GHz]
TMz014: 1.448201 [GHz]

TEz111: 0.8500555 [GHz]
TEz112: 0.9507808 [GHz]
TEz113: 1.098317 [GHz]
TEz114: 1.276536 [GHz]
```

Step 2. Two Level Preconditioner

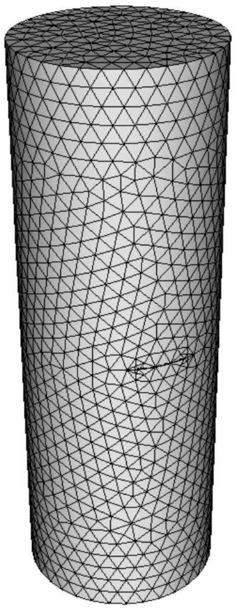
- Convergence Behavior compared to PCA tolerance



- IBC
- Freq: 1132.4207 MHz
- # of Basis: 28,944 (~ $\lambda / 15$ Mesh)
- The finest block size: 0.5λ
- The number of oct-tree level: 4
- EFIE + IEDG (with MLFMM)
- Iterative Solver (GCR, $\varepsilon_k = 10^{-6}$)
- Approximate Direct Inverse
  P_6: Schur-PCA ($\epsilon = 10^{-4}$)
  Q_k: $(P_1 A)^{-1}$ with PCA ($k = 50$)

Step 2. Two Level Preconditioner

- Memory Comparison



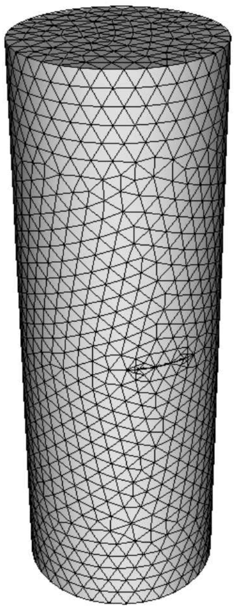
- IBC
- Freq: 1132.4207 MHz
- The finest block size: 0.5λ

	Block-Diag.	Approx. Direct Inverse	
Far Terms (MLFMM)	0.35	0.35	
Near/Self Terms	1.77	1.77	
Preconditioner	0.33	P_6	2.57
		Q_k	0.04
Total Memory	2.45	4.73	

Table 1. Memory Comparison [GB]

Step 2. Two Level Preconditioner

➢ Wall Time Comparison



- IBC
- Freq: 1132.4207 MHz
- The finest block size: 0.5λ

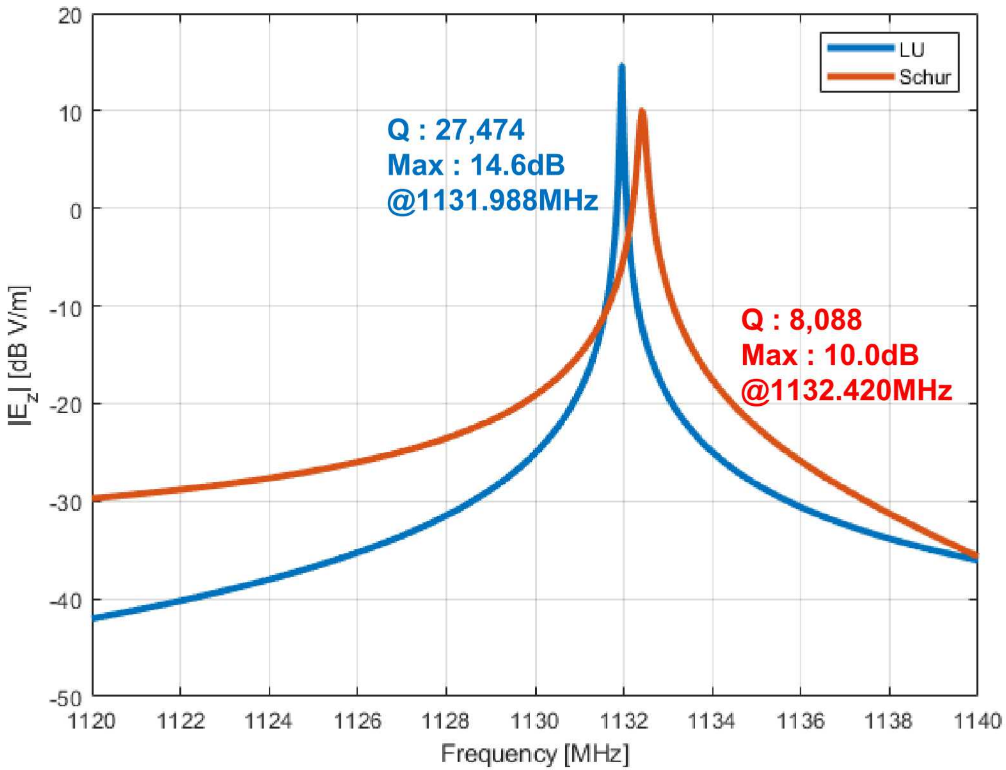
<table>
<tr><th></th><th>Block-Diag.</th><th colspan="2">Approx. Direct Inverse</th></tr>
<tr><td>Filling Far Terms (MLFMM)</td><td>1.2</td><td colspan="2">1.2</td></tr>
<tr><td>Filling Near/Self Terms</td><td>642.3</td><td colspan="2">642.3</td></tr>
<tr><td rowspan="2">Preconditioner</td><td rowspan="2">0.5</td><td>P_6</td><td>156.8</td></tr>
<tr><td>Q_k</td><td>5.2</td></tr>
<tr><td>Solve System Matrix</td><td>725.6</td><td colspan="2">11.3</td></tr>
<tr><td>Total Wall Time</td><td>1369.6</td><td colspan="2">816.8</td></tr>
</table>

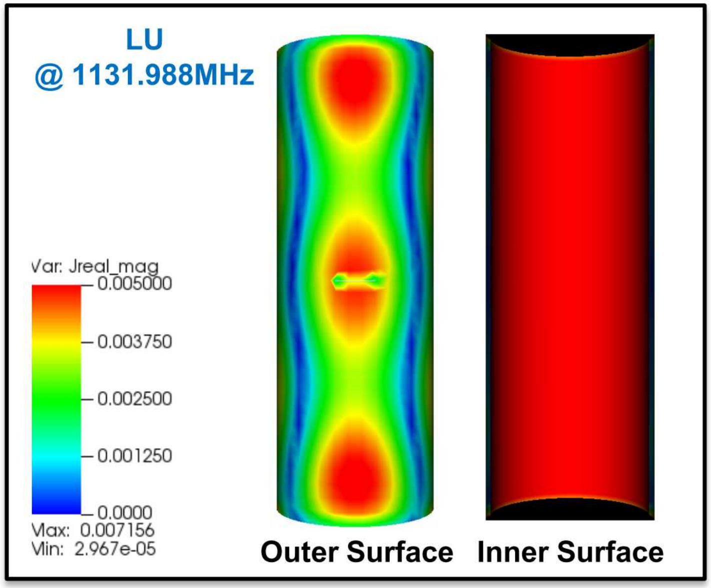
Table 2. Wall Time Comparison [sec]

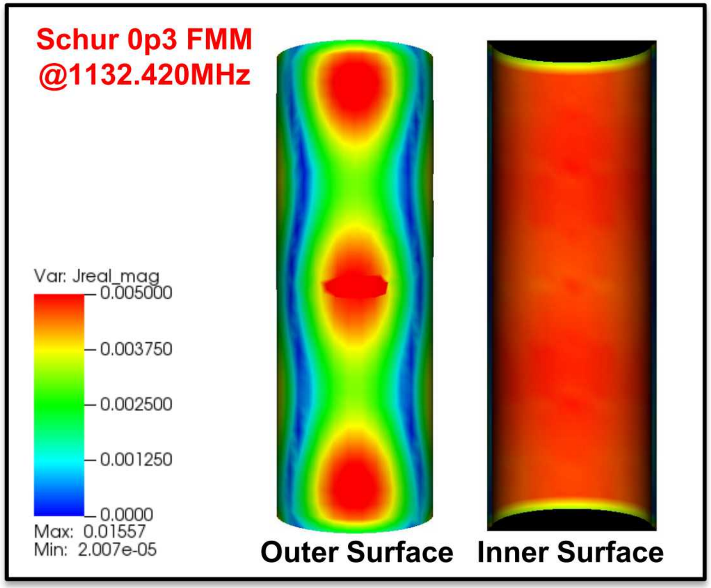
THE OHIO STATE UNIVERSITY

Step 2. Two Level Preconditioner

- Shielding Effectiveness

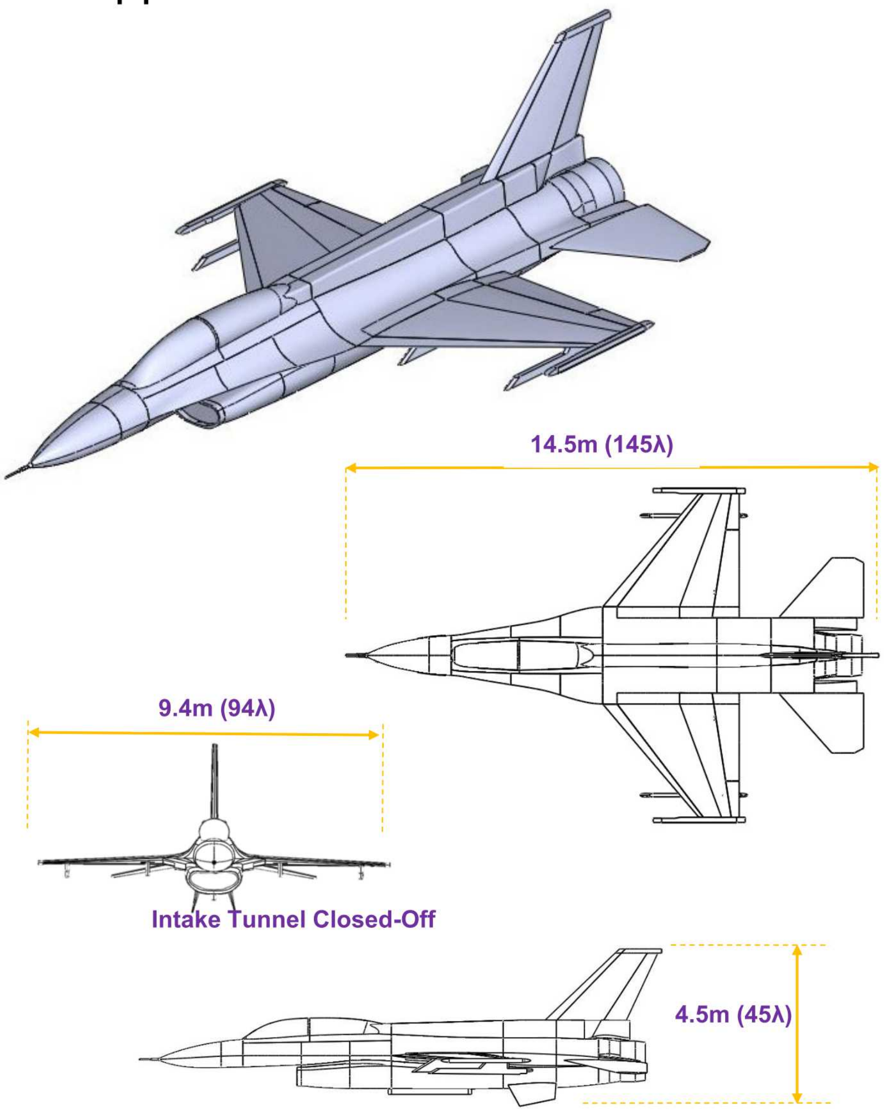






Step 2. Two Level Preconditioner

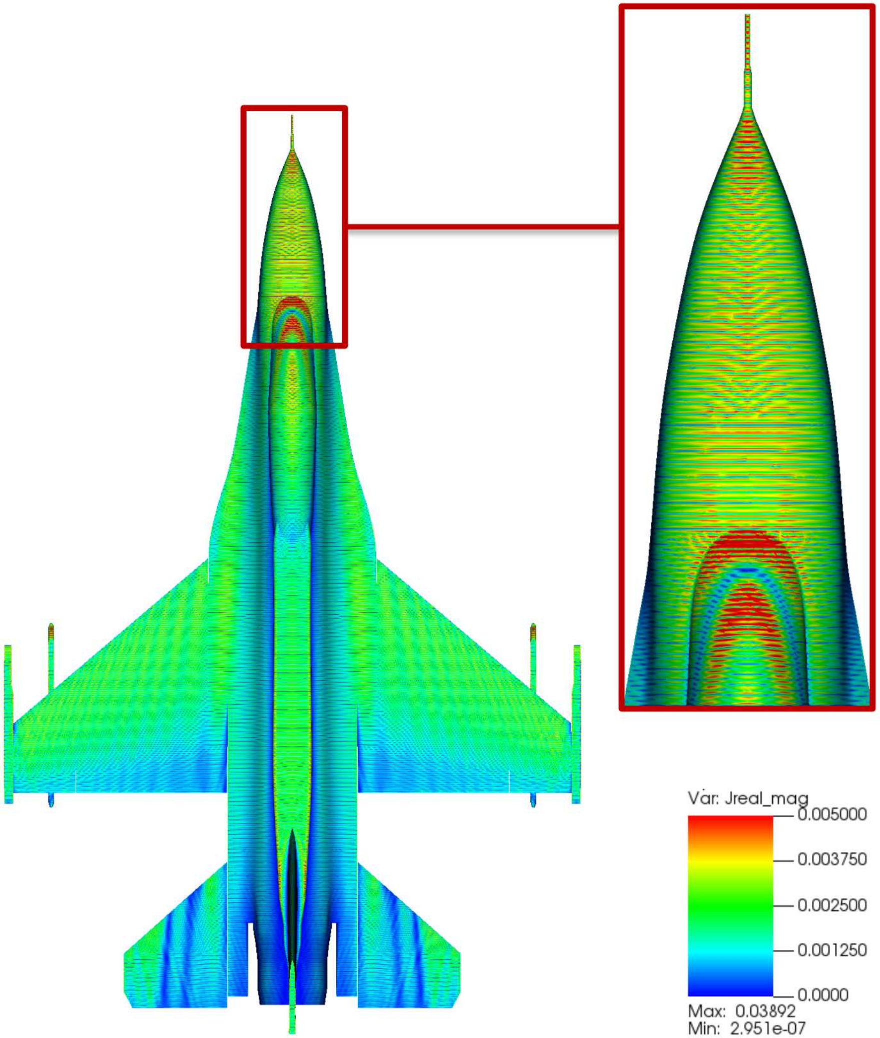
- Application: PEC F-16

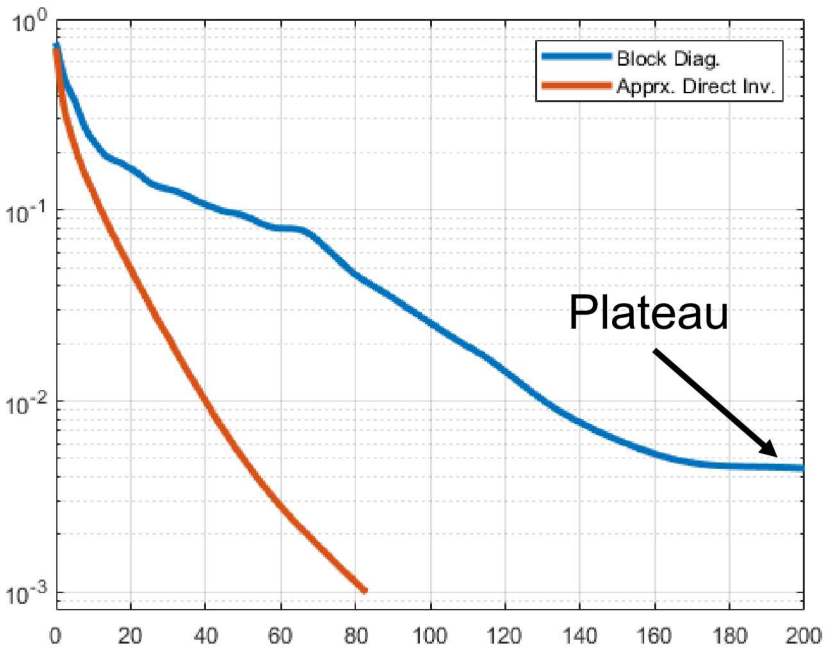


- 5GHz Head on Incident
- # of Basis: 7,700,888 ($\lambda/5$ Mesh)
- The finest FMM block size: 0.5 λ
- The number of MLFMM level: 10
- CFIE + IEDG
- Iterative Solver (GCR, $\epsilon_r = 10^{-3}$)
- 2 x Intel Xeon 6148, 48 threads

Step 2. Two Level Preconditioner

- Application: PEC F-16





- Freq: 5GHz
- \# of Basis: 7,700,888 ($\lambda/5$ Mesh)
- The finest FMM block size: 0.5λ
- Number of Recycle: 40
- Approximate Direct Inverse
  P_6: Schur-PCA ($\epsilon = 10^{-4}$)
  $\mathrm{Q_k}$: $(P_1 A)^{-1}$ with PCA ($k = 50$)

Step 2. Two Level Preconditioner

- Application: PEC F-16

	Block-Diag.	Approx. Direct Inverse	
Far Terms (MLFMM)	93.1	93.1	
Near/Self Terms	61.4	61.4	
Preconditioner	12.2	P_6	57.8
		Q_k	11.4
Total Memory	166.7	223.7	

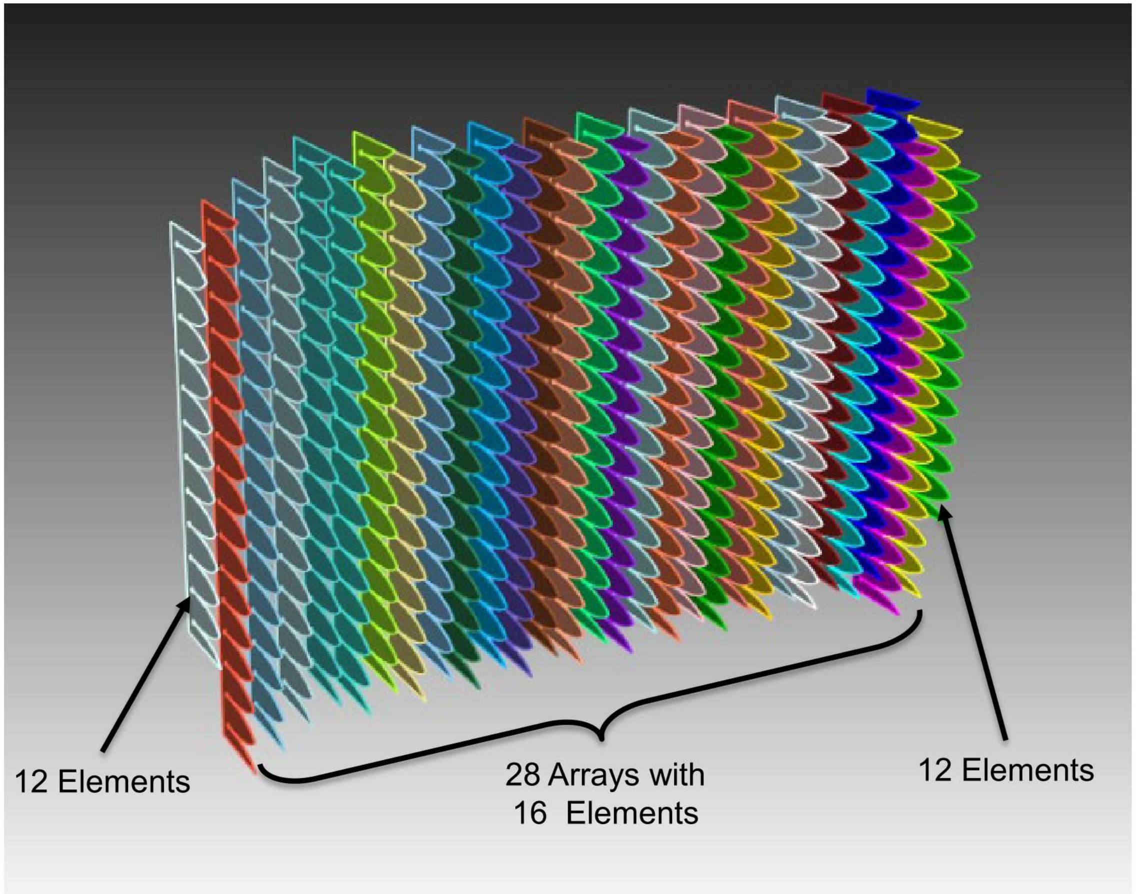
Table 1. Memory Comparison [GB]

	Block-Diag.	Approx. Direct Inverse	
Filling Far Terms (MLFMM)	4.7	4.7	
Filling Near/Self Terms	56.9	56.9	
Preconditioner	0.5	P_6	16.8
		Q_k	8.1
Solve System Matrix	38.6 (Plateau)	17.8	
Total Wall Time	100.7	104.3	

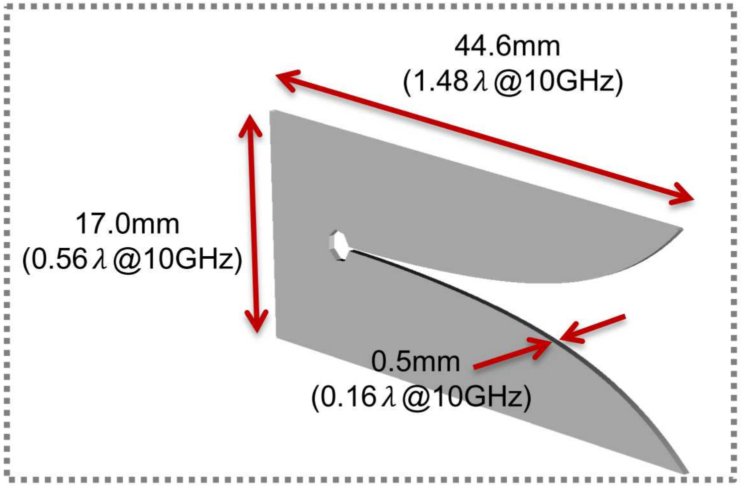
Table 2. Wall Time Comparison [min]

Step 2. Two Level Preconditioner

- Application: Monostatic RCS of PEC Vivaldi Antenna



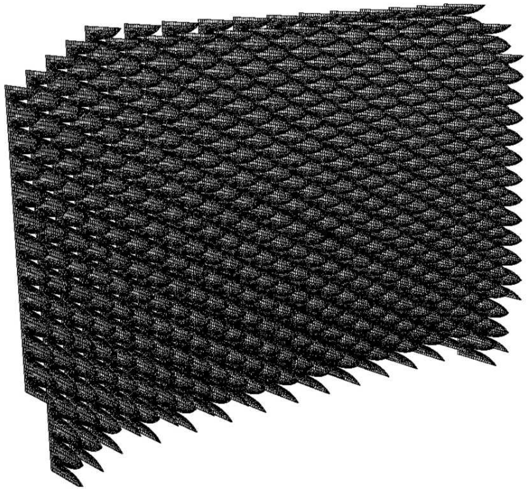
472 Elements Vivaldi Antenna

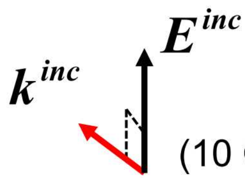


Unit Elements

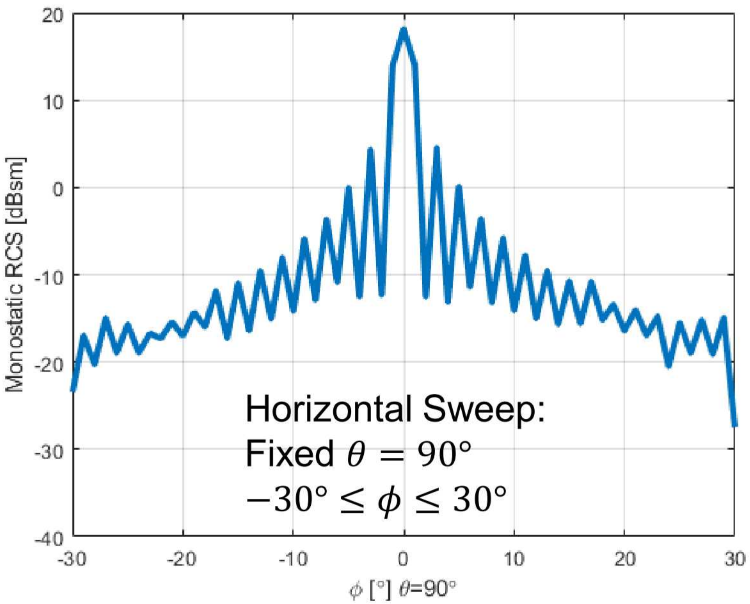
Step 2. Two Level Preconditioner

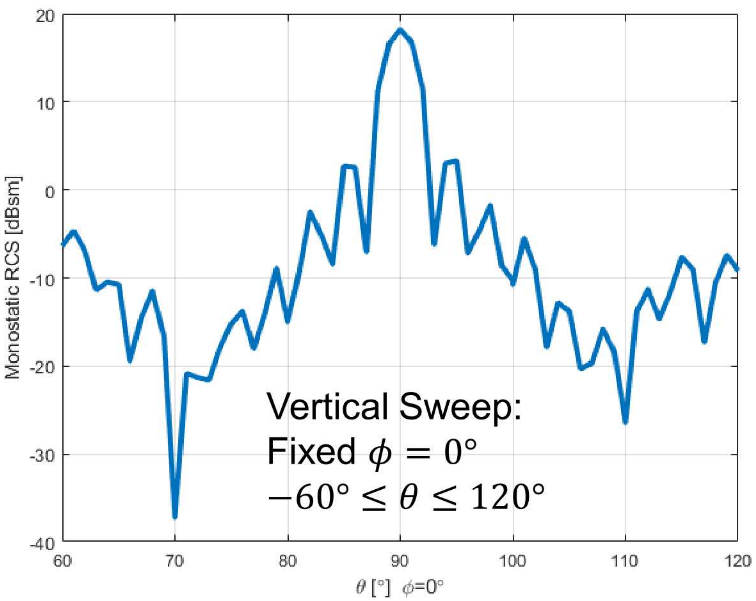
➢ Application: Monostatic RCS of PEC Vivaldi Antenna





- # of Basis: 2,350,240 ($\sim\lambda/25$ Mesh)
- The finest FMM block size: 0.5λ
- The number of MLFMM level: 6
- CFIE + IEDG
- Iterative Solver (GCR, $\varepsilon_k = 10^{-2}$)

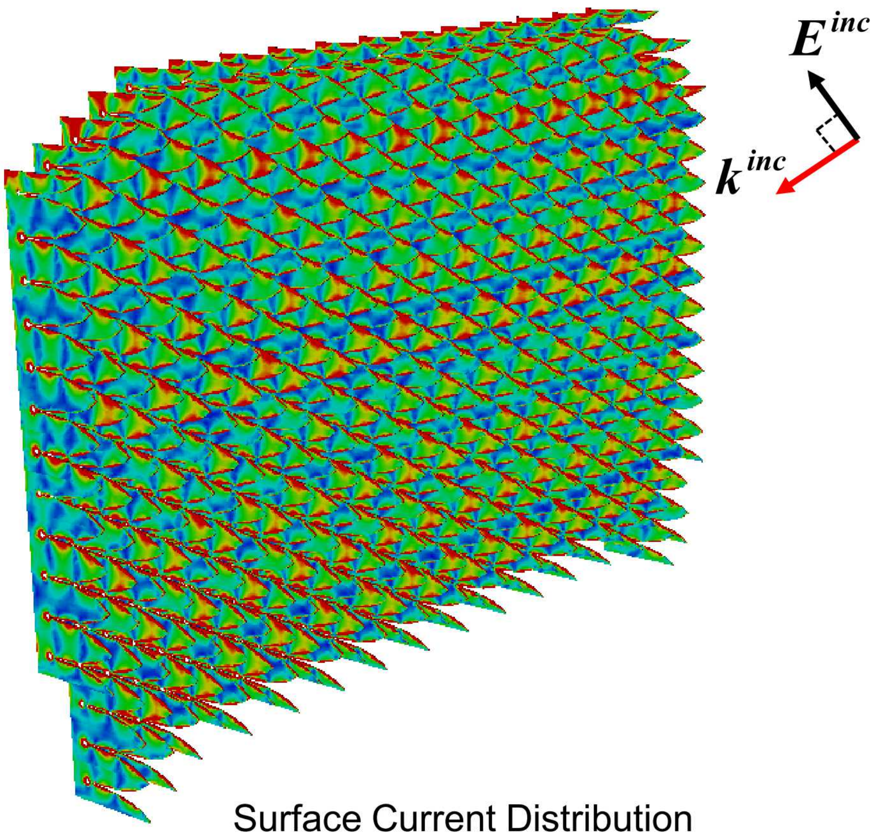




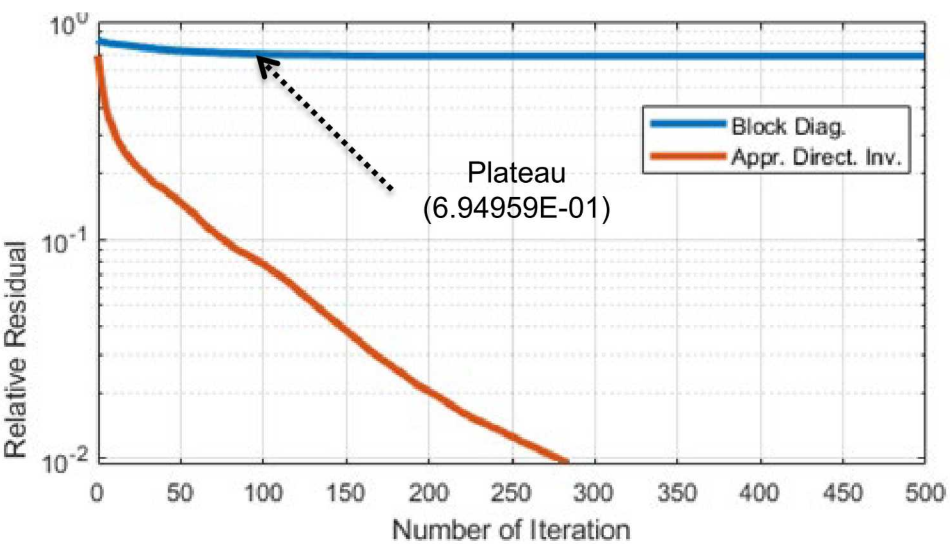
Step 2. Two Level Preconditioner

➢ Single Incident $(10\text{GHz}, \theta = 60°, \phi = 0°)$

*# of Recycle: 40



Surface Current Distribution



Convergence Behavior

Step 2. Two Level Preconditioner

- Computation Resources

	Memory [GB]	Wall Time [min]
Far Terms (MLFMM)	21.7	1.0
Near/Self Terms	89.9	185.2
Preconditioner	84.6	41.6
61 RHS Iterative Solve	22.4	574
Total	218.6	801.8

THE OHIO STATE UNIVERSITY

Thank You

Sandia National Laboratories

Acknowledgement: 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