Fast Algorithm for Matrix-Vector Multiply of Asymmetric Multilevel Block-Toeplitz Matrices

B. E. Barrowes\(^1\), F. L. Teixeira\(^2\), and J. A. Kong\(^1\)

\(^1\)Research Laboratory of Electronics, MIT, Cambridge, MA 02139-4307
\(^2\)ElectroScience Laboratory and Department of Electrical Engineering, The Ohio State University, Columbus, OH 43210
Presentation Outline

I. Introduction and Motivation

II. Definitions and Formulation

III. Recursive Algorithms

IV. Examples
   - Scattering from arbitrary object
   - 3-D SMCG

V. Conclusion
Introduction and Motivation

Multilevel block-Toeplitz (MBT) matrices often arise in the solution of electromagnetic scattering problems due to the translational invariance and convolutional nature of the Green’s function.

- Discrete Dipole Approximation (DDA)
- 2-D and 3-D scattering based on cubic meshes
- Other situations, e.g. autocorrelation of 2-D random process

Conjugant Gradient (CG) solvers are based on matrix-vector multiplies
- $O(N^2)$ complexity
- $O(N^2)$ memory storage

MBT Fast multiply reduces these requirements
- $O(N \log N)$ complexity
- $O(N)$ memory storage
- 1-D FFT
Problem Formulation

For 1-D point interactions:

Interaction matrix is Toeplitz

\[
\begin{bmatrix}
G_0 & G_1 & G_2 & G_3 \\
G_{-1} & G_0 & G_1 & G_2 \\
G_{-2} & G_{-1} & G_0 & G_1 \\
G_{-3} & G_{-2} & G_{-1} & G_0 \\
\end{bmatrix}
\]
Definitions

Let $A$ be a square matrix in the matrix equation $A \cdot x = b$

$$A = \begin{bmatrix} a_0^1 & a_1^1 & \cdots & a_{n_1}^1 \\ a_0^1 & a_0^1 & \cdots & a_{n_1}^1 \\ \vdots & \ddots & \ddots & \vdots \\ a_{n_1}^1 & \cdots & \cdots & a_0^1 \\ a_{n_1}^1 & \cdots & \cdots & a_{n_1}^1 \\ \end{bmatrix} , \quad \text{where}$$

$$a_{n_1}^1, \ldots, a_{n_1}^1 = \begin{bmatrix} a_0^2 & a_1^2 & \cdots & a_{n_2}^2 \\ a_0^1 & a_0^2 & \cdots & a_{n_2}^2 \\ \vdots & \ddots & \ddots & \vdots \\ a_{n_2}^2 & \cdots & \cdots & a_0^2 \\ a_{n_2}^2 & \cdots & \cdots & a_{n_2}^2 \\ \end{bmatrix} ,$$

continuing until the $(M - 1)^{st}$ level

$$a_{n(M-1)}^{M-1}, \ldots, a_{n(M-1)}^{M-1} = \begin{bmatrix} a_0^M & a_1^M & \cdots & a_{n_M}^M \\ a_0^1 & a_0^M & \cdots & a_{n_M}^M \\ \vdots & \ddots & \ddots & \vdots \\ a_{n_M}^M & \cdots & \cdots & a_0^M \\ a_{n_M}^M & \cdots & \cdots & a_{n_M}^M \\ \end{bmatrix} ,$$

with $a_{n_M}^{M}, \ldots, a_{n_M}^{M}$ arbitrary $f \times f$ matrices.

$$T_j^M(n_1, n_2, \ldots, n_M) \quad \text{or simply} \quad T_j^M$$

Example of a $A = T_1^3(2, 2, 2)$ asymmetric MBT matrix. Unique entries are boxed with a solid line. In the upper left hand corner, $a_0^1$ is enclosed by a dash-dot box, while at the lower left hand corner $a_{n_1}^1a_{n_2}^1$ in delineated by a dashed box. Thin solid lines delineate interaction between different rows of a 2x2x2 cubic mesh, while bold solid line delineate between planes.
Fast Multiply – Recursive Algorithms

Fast FFT assisted multiply requires 4 steps

1. Assign unique entries of $A$ to $A_u$
   $$A_u = \Pi_{bl}(A)$$

2. Insert zeros into $x$
   $$x_z = \zeta_p(x)$$

3. FFT of $A_u$ and $x_z$, multiplication in the Fourier domain
   $$\widetilde{A}_u, \, \widetilde{x}_z \leftrightarrow A_u, \, x_z$$

4. Reconstruct $b$ from $b_z$
   $$b = \zeta_a^{-1}(b_z)$$
1. Assign unique entries of $A$ to $A_u$ \[ A_u = \Pi_{bt}(A) \]

Proceed recursively from lower left to upper right

Final $f \times f$ block’s order is:

\[
\begin{bmatrix}
  f_1 & f_2 & f_3 \\
  f_4 & f_5 & f_6 \\
  f_7 & f_8 & f_9
\end{bmatrix} \rightarrow [f_7, f_8, f_9, f_4, f_5, f_6, f_1, f_2, f_3].
\]

For $A$ above, $A_u = \Pi_{bt}(A) = [1 : 27]$

Because only $A$ is stored, memory requirement is reduced from

\[
f^2 \left( \prod_{i=1}^{M} n_i \right)^2 = f^2 N^2
\]

to

\[
f^2 \prod_{i=1}^{M} (2n_i - 1) \approx f^2 2^M N
\]
2. Insert zeros into $\mathbf{x}$ \quad $\mathbf{x}_z = \zeta_p(\mathbf{x})$

Given $n_1, n_2, \ldots, n_M$ and $f$, proceed recursively

1. Insert 0's for unique entries along LHS of Toeplitz blocks
2. Zero pad to length of $\mathbf{A}_u$

For $\mathbf{x} = [1:8]$, $\mathbf{x}_z = \zeta_p(\mathbf{x}) = [1, 2, 0, 3, 4, 0, 0, 0, 0, 5, 6, 0, 7, 8, 0, 0, 0, 0|0, 0, 0, 0, 0, 0, 0, 0]$
Then

\[ A_u = \Pi_{ht}(A) = [1 : 27] \]

\[ x_z = \zeta_p(x) = \text{(see below)} \]
3. Fast Fourier Transforms

Now convolution becomes a multiplication in Fourier domain

1. FFT of $A_u$ and $x_z$, multiplication in the Fourier domain
   $$\tilde{A}_u \cdot \tilde{x}_z \overset{\mathcal{F}}{\leftrightarrow} A_u \otimes x_z$$

For CG type solvers, perform $\mathcal{F}(A_u)$ only once

Therefore only 2 FFTs required per iteration
4. Reconstruct $b$ from $b_z$ — $b = \zeta_a^{-1}(b_z)$

Some terms (corresponding to inserted zeros) must be removed from $b_z$

For $A$ and $x$ above,

$$
\begin{align*}
    b &= \zeta_a^{-1}(b_z) \\
    &= \zeta_a^{-1}(\mathcal{F}^{-1}(\widetilde{A}_u \cdot \widetilde{x}_z)) \\
    &= \zeta_a^{-1}\left[\mathcal{F}^{-1}\left(\Pi_{bt}(\widetilde{A}) \cdot \widetilde{\zeta_p}(x)\right)\right] \\
    &= [824, 788, 716, 680, 500, 464, 392, 356]
\end{align*}
$$

For further detail and sample Matlab codes, see Barrowes et al., MOTL, to appear October 2001.
Example I

Scattering from an Arbitrarily Shaped Object

Small dielectric spheres placed on lattice point (similar to DDA)

- Scattering from 1 small particle

\[ \overline{E}_s(\mathbf{r}) = \frac{-k^2}{\epsilon} \overline{G}(\mathbf{r}, \mathbf{r}') v_0(\epsilon_p - \epsilon) \overline{E}(\mathbf{r}') , \]

- Self-consistent solution for scattering from many small particles

\[ \overline{E}(\mathbf{r}_i) = \overline{E}_{\text{inc}}(\mathbf{r}_i) + \frac{k^2}{\epsilon} \sum_{j=1}^{N} v_j (\epsilon_j - \epsilon) \overline{G}(\mathbf{r}_i, \mathbf{r}_j) \overline{E}(\mathbf{r}_j) . \]

- For otherwise similar spheres on lattice points, the interaction between particles \( i \) and \( j \) can be seen to depend only on \( \overline{G}(\mathbf{r}_i, \mathbf{r}_j) \) which is translationally invariant.

- For 3-D, this creates a \( T^M_j(N_x, N_y, N_z) \)
Example I Results

<table>
<thead>
<tr>
<th>( N_q ) (FFT length)</th>
<th>MBT Fast Method Time (s)</th>
<th>Full Method Time (s)</th>
<th>%error</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>19773</td>
<td>1.33</td>
<td>5.85</td>
</tr>
<tr>
<td>10</td>
<td>61731</td>
<td>5.98</td>
<td>25.5</td>
</tr>
<tr>
<td>15</td>
<td>219501</td>
<td>19.25</td>
<td>190.7</td>
</tr>
<tr>
<td>23</td>
<td>820125</td>
<td>87.9</td>
<td>2544.0</td>
</tr>
<tr>
<td>31</td>
<td>2042829</td>
<td>363.3</td>
<td>14295.3</td>
</tr>
</tbody>
</table>

3D diagram of a spherical object.
Example I Results

<table>
<thead>
<tr>
<th>$N_g$</th>
<th>$N$(cube)</th>
<th>$\epsilon_{\text{eff}}$</th>
<th>$%$error</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>343</td>
<td>1.7688+0.0266i</td>
<td>0.0</td>
</tr>
<tr>
<td>10</td>
<td>1000</td>
<td>1.7867+0.007321</td>
<td>23.5</td>
</tr>
<tr>
<td>15</td>
<td>3375</td>
<td>1.7457+0.017031</td>
<td>13.7</td>
</tr>
<tr>
<td>23</td>
<td>12167</td>
<td>1.7517+0.018341</td>
<td>2.997</td>
</tr>
<tr>
<td>31</td>
<td>29791</td>
<td>1.7471+0.018291</td>
<td>3.01</td>
</tr>
</tbody>
</table>

$\text{Scattering Angle} - \theta$

$\text{Scattering Intensity}$
Example II – 3-D SMCG for Spheroids
Example II Results

<table>
<thead>
<tr>
<th>Expansion order ($\gamma$)</th>
<th>Number of expansion terms ($G_{\gamma x \gamma y \gamma z}$)</th>
<th>$N_T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>70</td>
</tr>
<tr>
<td>5</td>
<td>21</td>
<td>112</td>
</tr>
</tbody>
</table>

Number of expansion terms and total number of FFTs (both forward and inverse) for expansion order $\gamma$.

Computation times and Kullback-Leibler distances for SMCG method for $\gamma=0.5$ with $r_d=1.1 \Delta r_g$ and $N_{cube}=10000$. 
Conclusion

A new $O(N \log N)$ FFT-based method to expedite matrix-vector multiplies involving multilevel block-Toeplitz (MBT) matrices was presented.

- This method applies to MBT matrices with any $M$
- This is a minimal memory method storing only the unique entries of $A$
- This method is applicable to many electromagnetic problems based on cubic meshes
- This method was shown to apply to scattering from an object of arbitrary shape and shown useful in an extension of the SMCG method to 3-D.

This work was supported by National Science Foundation (NSF) under grant numbers ECS9615799 and ECS9423861, by the Office of Naval Research (ONR) under contract numbers N00014-99-1-0175 and N00014-97-1-0172, and through a NSF Graduate Fellowship.