

Fast Multiplication of Matrix-Vector by Virtual Grids Technique in AIM

Mingxuan Zheng*, Huiling Zhao, and Zhonghui Zhao

Abstract—In order to accelerate the speed of matrix-vector product (MVP) in iteration process for adaptive integral method (AIM), a virtual grids technique (VGT) with multi-dimensional fast Fourier transform (FFT) is proposed. By adding some uniform virtual grids outside the original region, the indexes of nonzeros in the projection matrix are modified so as to eliminate the padding and unpadding procedures in MVP. The advantages of this method are that first it will not occupy any extra memory, and second it makes the Green's function vector compressed from $(2N_x - 1)(2N_y - 1)(2N_z - 1)$ to $8(N_x - 1)(N_y - 1)(N_z - 1)$ because of its symmetrical block-Toeplitz property. Numerical results show that per iteration time could be reduced more than 30% by applying this method in comparison with the conventional AIM, without losing accuracy. In addition, the peak memory consumption could also be reduced because the intermediate vectors are eliminated.

1. INTRODUCTION

In the past decades, many fast methods, based on method of moment (MoM) [1], have been proposed to solve large-scale electromagnetic problems, such as the multilevel fast multipole algorithm (MLFMA) [2, 3], AIM [4–7], and its close counterpart, the precorrected fast Fourier transformation (P-FFT) [8]. MLFMA and AIM both can reduce the computational complexity from $O(N^2)$ to $O(N \log(N))$, and the memory requirements from $O(N^2)$ to $O(N)$. However, AIM is less dependent on the integral kernels than MLFMA. This feature makes it easier to implement with the standard and efficient fast Fourier transform (FFT) libraries available online.

In AIM, to accelerate MVP during each iteration by one-dimensional (1D) FFT [9], the block-Toeplitz matrix on auxiliary grids must be formulated as a circular Toeplitz matrix. This suggests that the vector should be padding zeros before FFT and unpadding zeros after inverse fast Fourier transform (IFFT). However, the strategies of the two procedures are rather complicated, which would consume a lot of extra CPU instructions to copy data. In addition, some temporary vectors would be created explicitly to keep the intermediate results. Consequently, the speed of MVP would be slowed down severely when more and more auxiliary grids are required to solve the large-scale problems via an iterative solver.

To overcome these drawbacks, in this paper, a virtual grids technique is proposed to accelerate the MVP with the help of multi-dimensional FFT. Different from the 1D FFT in [9], the proposed method would eliminate the procedures of padding and unpadding zeros, which could save not only time but also the temporary memory during MVP. Moreover, compared with the traditional multi-dimensional FFT involved in AIM [4], the length of the circular vector is $8(N_x - 1)(N_y - 1)(N_z - 1)$ for 3D problems when electric field integral equation (EFIE) is used, which is slightly less than the one required in the traditional AIM $(2N_x - 1)(2N_y - 1)(2N_z - 1)$, where N_x , N_y , N_z are the numbers of grids in x , y , z directions, respectively. Furthermore, without much modification, the proposed method is friendly to

Received 8 November 2019, Accepted 7 January 2020, Scheduled 22 January 2020

* Corresponding author: Mingxuan Zheng (x408859786@mail.nwpu.edu.cn).

The authors are with the School of Electronics and Information, Northwestern Polytechnical University, Xi'an 710129, China.

incorporate into the existing code, because it is independent of basis functions, integral kernels, and the solver.

The rest of this paper is organized as follows. Section 2 briefly reviews AIM and the MVP by 1D FFT. In Section 3 the virtual grids technique with multi-dimensional FFT is detailed. Some numerical results are given in Section 4 to validate the accuracy and efficiency of the proposed method, and Section 5 concludes this paper.

2. FORMULATIONS

For an arbitrary 3D perfect electric conductor (PEC) structure, the conventional MoM is applied to form a linear system as

$$\mathbf{Z}\mathbf{I} = \mathbf{V} \quad (1)$$

where \mathbf{I} is the current coefficients of basis functions, \mathbf{V} the excitation, and \mathbf{Z} the impedance matrix. To reduce the memory requirements of \mathbf{Z} and computational complexity in solving Eq. (1) with an iterative solver, AIM divides the impedance matrix into near and far matrices as $\mathbf{Z} = \mathbf{Z}^{near} + \mathbf{Z}^{far}$. The detailed definition of \mathbf{Z}^{near} could be found in [4], and only \mathbf{Z}^{far} is compressed as the multiplications of several sparse matrices

$$\mathbf{Z}^{far} = \sum_{u=x,y,z,d} \Lambda_u \mathbf{G} \Lambda_u^T \quad (2)$$

where \mathbf{G} is the block-Toeplitz transformation matrix on the auxiliary grids, and Λ is the sparse projection matrix to construct a relationship between basis functions and global auxiliary grids. The order of \mathbf{G} is the number of all the grids $N_C = N_x \times N_y \times N_z$, and the size of Λ is $N \times N_C$, where N is the number of the basis functions. Then, the MVP in AIM could be written as

$$\mathbf{I}_{i+1} = \mathbf{Z}^{near} \mathbf{I}_i + \sum_{u=x,y,z,d} \Lambda_u \text{IFFT}\{\text{FFT}[\mathbf{G}] \circ \text{FFT}[\Lambda_u^T \mathbf{I}_i]\} \quad (3)$$

where \circ denotes the Hadamard product operator. In fact, it is impossible to save the whole \mathbf{G} since $N_C \gg N$. Instead, the block-Toeplitz \mathbf{G} is saved as $\tilde{\mathbf{G}} = \text{FFT}[\mathbf{G}]$. As a result, for 3D electromagnetic problems, it is necessary to transform \mathbf{G} into a circular Toeplitz matrix before FFT, which means that the total length of sequences $\tilde{\mathbf{G}}$ is increased from N_C to $N_{FFT} = (2N_x - 1)(2N_y - 1)(2N_z - 1)$. Fig. 1(a) illustrates the details of MVP for the second term on the right side of Eq. (3) by 1D FFT [9]. Firstly, the intermediate vector is derived by $\mathbf{V}_{N_C} = \Lambda^T \mathbf{I}$. To realize the procedure of padding zeros, a new vector $\mathbf{V}_{N_{FFT}}$ with length N_{FFT} is initialized by zeros, and the complex values are copied from \mathbf{V}_{N_C} to the appropriate positions of $\mathbf{V}_{N_{FFT}}$. Then $\mathbf{V}_{N_{FFT}}$ is operated by FFT, Hadamard product with $\tilde{\mathbf{G}}$ and IFFT. Finally, the useful values are extracted from $\mathbf{V}_{N_{FFT}}$ and written back into \mathbf{V}_{N_C} through unpadding procedure. To finish MVP, \mathbf{I} is rewritten by the multiplication of Λ and \mathbf{V}_{N_C} .

In the above method, there are two problems that influence the performance of MVP. The first one is the copy operations between $\mathbf{V}_{N_{FFT}}$ and \mathbf{V}_{N_C} involved in padding and unpadding procedures. The second one is the requirement of a temporary vector \mathbf{V}_{N_C} for keeping the intermediate results. When N_C is increased, the former will slow down the speed of MVP, and the later will increase the peak memory requirements during iteration. To reduce both the computational complexity and memory requirements without any extra memory, the virtual grids technique is introduced in the next section.

3. VIRTUAL GRIDS TECHNIQUE

Before introducing the virtual grids technique, it is important to illustrate the transformation of \mathbf{G} firstly. To meet the circular property of the block-Toeplitz, the length of Green's function vector \mathbf{g} should be enlarged from N_C to $N_{FFT} = (2N_x - 1)(2N_y - 1)(2N_z - 1)$ before FFT. Due to the symmetry of \mathbf{G} when EFIE is applied, i.e., $\mathbf{G}_{1,N_x,y,z}$ is equal to $\mathbf{G}_{N_x,y,z,1}$, then the length of the vector \mathbf{g}' could be slightly reduced from N_{FFT} to $N'_{FFT} = 8(N_x - 1)(N_y - 1)(N_z - 1)$. Two examples of 1D ($N_x = 4$)

and 2D ($N_x = N_y = 3$) Green's function vectors are given as

$$1D: \mathbf{g} = [g_1 \ g_2 \ g_3 \ g_4]^T \text{ to } \mathbf{g}' = [g_1 \ g_2 \ g_3 \ g_4 \ g_3 \ g_2]^T \quad (4)$$

$$2D: \mathbf{g} = \begin{pmatrix} g_1 & g_4 & g_7 \\ g_2 & g_5 & g_8 \\ g_3 & g_6 & g_9 \end{pmatrix} \text{ to } \mathbf{g}' = \begin{pmatrix} g_1 & g_4 & g_7 & g_4 \\ g_2 & g_5 & g_8 & g_5 \\ g_3 & g_6 & g_9 & g_6 \\ g_2 & g_5 & g_8 & g_5 \end{pmatrix} \quad (5)$$

where \mathbf{g} and \mathbf{g}' are the first row of original and circular block-Toeplitz matrix. It is worthy to note that \mathbf{g}' is saved column by column in 2D cases. Compared with the original vector \mathbf{g} with grids of $N_x \times N_y \times N_z$, the modified \mathbf{g}' is just copying $N_x - 2$, $N_y - 2$, $N_z - 2$ values from \mathbf{g} in each direction.

To eliminate the procedures of padding and unpadding in Fig. 1(a), the virtual grids technique is designed to perform the two procedures implicitly through a modified projection matrix Λ_m . The complicated strategies are moved into Λ_m through changing the indexes of nonzeros. The intermediated vector \mathbf{V}_{N_C} is not required any more, and then the copy operations would also be eliminated. For simplicity, as shown in Fig. 2, a 2D scatterer is surrounded by the auxiliary grids denoted by circles with $N_x = 8$, $N_y = 5$. The size of Λ is $N \times N_C$, where $N_C = 40$. Then the virtual grids represented by crosses are added outside the original region. As a result, the whole region becomes larger than before, so that the size of Λ_m becomes $N \times N'_{FFT}$, where $N'_{FFT} = 4(N_x - 1)(N_y - 1) = 112$. This indicates that the indexes of the nonzeros should be changed to fit the new region.

Fortunately, the coefficients of these virtual grids in Λ_m are all zeros since none of the basis functions are expanded on the virtual grids. It seems that the so-called virtual grids do not exist for

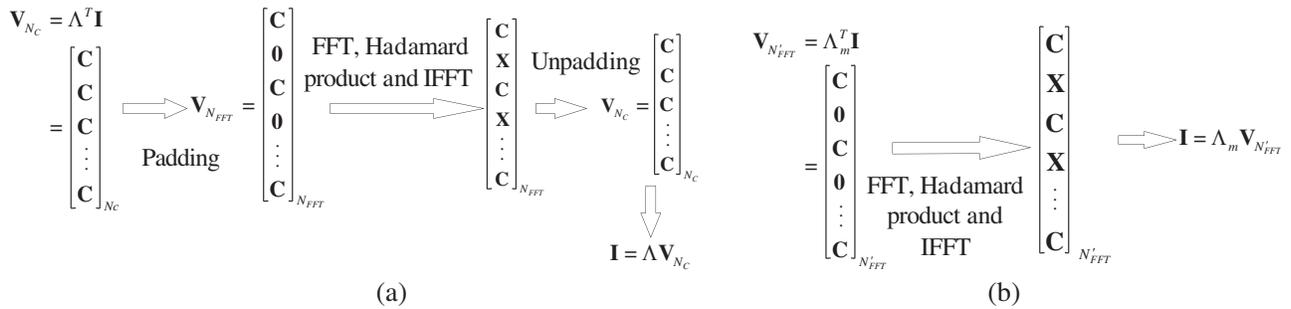


Figure 1. (a) 1D FFT and (b) n -dimensional FFT with virtual grids technique in MVP. The block vector \mathbf{C} denote the useful values and \mathbf{X} are ignored during unpadding.

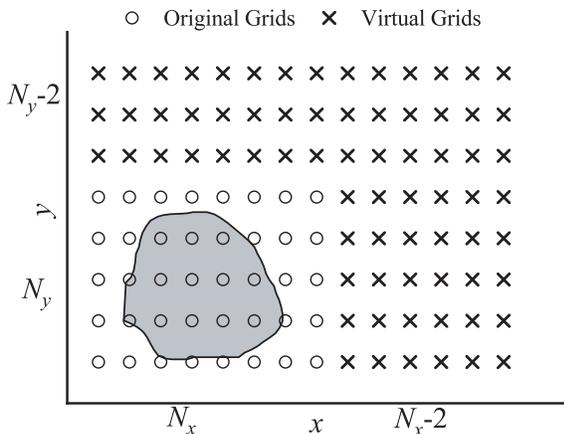


Figure 2. The original grids and extra virtual grids ($N_x = 8$, $N_y = 5$), where the PEC scatterer is depicted in gray.

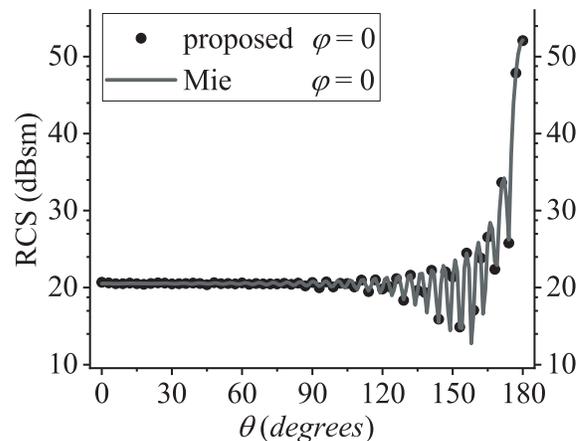


Figure 3. Comparison of the bistatic RCS by Mie series and the proposed method for a conducting sphere of radius 6λ .

Λ_m , which means that the memory requirements and the computational complexity of Λ_m are not increased compared to Λ . Only the indexes of nonzeros in Λ_m and Λ are different.

Next, as shown in Fig. 1(b), with the help of virtual grids technique, zeros are padded automatically in the correct positions of $\mathbf{V}_{N'_{FFT}}$ by the multiplication of Λ_m^T and \mathbf{I} . The copy operations are saved for the first time. After FFT, Hadamard product with $\tilde{\mathbf{G}}$ and IFFT, $\mathbf{V}_{N'_{FFT}}$ is multiplied by Λ_m to extract useful values \mathbf{C} implicitly, then these values are written back into \mathbf{I} to finish MVP. There is no contribution from block vector \mathbf{X} since the coefficients in Λ_m related to \mathbf{X} are all zeros. This is the second time to eliminate the copy operations. On the other hand, compared with the method in Fig. 1(a), the proposed method eliminates the temporary vector \mathbf{V}_{N_C} . It is easy to estimate that the computational complexity of copy operations and memory requirements of the temporary vector \mathbf{V}_{N_C} are $O(N_C)$, which would influence the performance of MVP dramatically when N_C is quite large.

4. NUMERICAL SIMULATIONS

To demonstrate the performance of the proposed method for the scattering problems, all the simulations are solved by the stabilized biconjugate gradient (Bi-CGSTAB) with the relative residual error less than 10^{-4} . The grid interval is 0.05λ ; the grid order is 2; and the near threshold is 0.3λ . All the numerical experiments are performed on the desktop PC of four cores 4790 Intel processors with math kernel library to calculate 1D and 3D FFT. The first and second examples are simulated to verify the accuracy of the proposed method, and the last one is to validate the performance of time and memory reduction in iterations.

The first example is a PEC sphere of radius 6λ illuminated by a plane wave from azimuthal angle $(\theta, \varphi) = (0, 0)$ with x -axis polarization. The number of unknowns of the sphere is 128,571, and the number of auxiliary grids is 14,348,907 which will occupy about 875 MB extra memory to keep four intermediated vectors V_{N_C} if the original MVP is applied. Fig. 3 shows that the proposed method agrees well with Mie series in the bistatic radar cross section (RCS).

To further verify the accuracy of the proposed method, the second example is a PEC hourglass located on the xoy plane illuminated by an incident plane wave from $(\theta, \varphi) = (\pi/2, 0)$ with z -axis polarization. As shown from left-top corner of Fig. 4(a), the top and bottom radii of the hourglass are 4λ ; the radius of the junction circle in the middle is 1λ ; and the height is 8λ , which is discretized by 140,661 basis functions. It can be seen from Fig. 4 that the bistatic RCSs of this hourglass in horizontal and vertical planes calculated by the proposed method agree well with MLFMA.

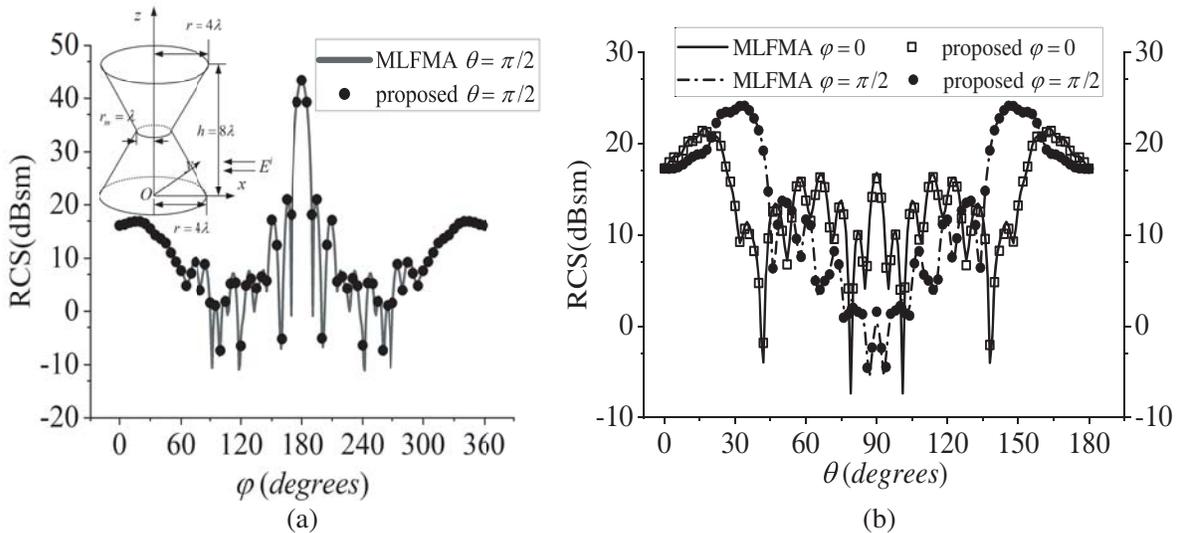


Figure 4. Comparison of RCS calculated by MLFMA and the proposed method for the hourglass in (a) horizontal, (b) vertical planes.

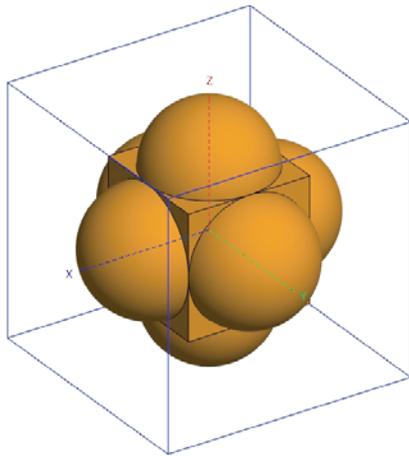


Figure 5. The module of the assembly of cubic and six half-spheres. The lengths of box are chosen as 2λ , 6.5λ and 10.5λ represented by CS2, CS6.5 and CS10.5 respectively.

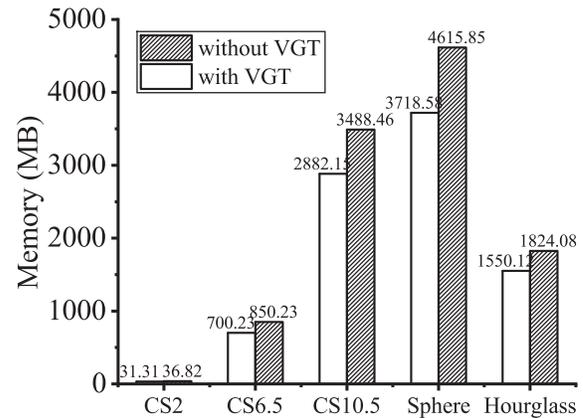


Figure 6. The comparison of the peak memory requirements during MVP with and without VGT.

The last example is an assembly of cubic and six half-spheres (CS) in different sizes as shown in Fig. 5. CS2, CS6.5, and CS10.5 denote the lengths of the box, which are 2λ , 6.5λ , and 10.5λ , respectively. To validate the performance of the proposed method in MVP, the plots of RCSs of these examples are omitted for redundancies. Table 1 summarizes the performance of CS in different sizes, the sphere, and hourglass with and without VGT in terms of per iteration CPU time. Obviously, the iteration time rises as the grid number increases. Compared with the far part, the time consumptions by the near part are less than 1% for all examples and can be ignored. The per iteration time of CS10.5 is slightly longer than sphere because there are more unknowns even though the grid number is lower. Compared with iteration time without VGT, the proposed method could reduce more than 30% time in all examples, especially for the hourglass, whose time is only 8.53 s but 20.47 s without VGT. On the other hand, the peak memory requirements of these examples during MVP are shown in the Fig. 6, which mainly include the consumptions of \mathbf{Z}^{near} , four projection matrices \mathbf{A} , $\tilde{\mathbf{G}}$ and the temporary vectors \mathbf{V}_{NC} for the original AIM. It can be seen from Fig. 6 that the memory requirements are rising as the grid number increases. The peak memory of all the examples without VGT is slightly higher than the ones with VGT, because the extra memory is consumed by the intermediated vector. This phenomenon becomes more and more apparent when the grid number is larger. In conclusion, without losing accuracy, the proposed method could reduce not only the time but also memory requirements during MVP.

Table 1. Iteration time for different examples.

Example	Grid number	Unknowns	Seconds per iteration			Time reduction
			Near part	Far part without VGT	Far part with VGT	
CS2	$43 \times 43 \times 43$	5 781	0.003	0.30	0.20	33%
CS6.5	$133 \times 133 \times 133$	61 062	0.026	8.89	4.48	49%
CS10.5	$213 \times 213 \times 213$	156 390	0.070	43.42	29.26	32%
Sphere	$243 \times 243 \times 243$	128 571	0.041	41.69	28.16	32%
Hourglass	$163 \times 163 \times 163$	140 661	0.064	20.40	8.46	58%

5. CONCLUSION

In this paper, the virtual grids technique combined with the multi-dimensional FFT is designed to accelerate the MVP during iteration in AIM. Through adding the virtual grids, the projection matrix is modified to eliminate the padding and unpadding procedures without any extra memory. The compressed Green's function vector in the proposed method is also slightly less than the one in the 1D FFT if EFIE is used. Moreover, since this method is independent of basis functions and integral kernel, it is easily incorporated into the existing code. The numerical results show that the proposed method agrees well with the analytical solution of sphere and the results of MLFMA. Compared with the original AIM, the per iteration time of the proposed method could save more than 30%, at most 58% for the example of hourglass. Also, the peak memory requirements is less than the one without VGT due to the elimination of the intermediated vector.

REFERENCES

1. Harrington, R. F., *Field Computation by Moment Methods*, Wiley-IEEE Press, New York, 1993.
2. Song, J., C. C. Lu, and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," *IEEE Transactions on Antennas and Propagation*, Vol. 45, No. 10, 1488–1493, October 1997.
3. Chew, W. C., T. J. Cui, and J. M. Song, "A FAFFA-MLFMA algorithm for electromagnetic scattering," *IEEE Transactions on Antennas and Propagation*, Vol. 50, No. 11, 1641–1649, January 2002.
4. Bleszynski, E., M. Bleszynski, and T. Jaroszewicz, "AIM: Adaptive integral method for solving large-scale electromagnetic scattering and radiation problems," *Radio Science*, Vol. 31, No. 5, 1225–1251, September 1996.
5. Yang, K. and A. E. Yilmaz, "An FFT-accelerated integral-equation solver for analyzing scattering in rectangular cavities," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 62, No. 9, 1930–1942, September 2014.
6. Wang, X., S. X. Gong, J. Ma, and C. F. Wang, "Efficient analysis of antennas mounted on large-scale complex platforms using hybrid AIM-PO technique," *IEEE Transactions on Antennas and Propagation*, Vol. 62, No. 3, 1517–1523, March 2014.
7. Wang, X., D. H. Werner, and J. P. Turpin, "Investigation of scattering properties of large-scale aperiodic tilings using a combination of the characteristic basis function and adaptive integral methods," *IEEE Transactions on Antennas and Propagation*, Vol. 61, No. 6, 3149–3160, June 2013.
8. Ling, F., V. I. Okhmatovski, W. Harris, S. McCracken, and A. Dengi, "Large-scale broad-band parasitic extraction for fast layout verification of 3-D RF and mixed-signal on-chip structures," *IEEE Transactions on Microwave Theory and Techniques*, Vol. 53, No. 1, 264–273, January 2005.
9. Barrowes, B. E., F. L. Teixeira, and J. A. Kong, "Fast algorithm for matrix-vector multiply of asymmetric multilevel block-toeplitz matrices," *IEEE Transactions on Antennas and Propagation Society International Symposium*, 630–633, Boston, USA, MA, July 2001.