The Array Decomposition-Fast Multipole Method

*R. W. Kindt[1] and J. L. Volakis[1,2]
[1]Radiation Lab., University of Michigan, Ann Arbor, MI 48109-2122
[2]ElectroScience Lab., Ohio State University, Columbus, OH 43212-1191

Abstract:

An innovative approach is presented for analyzing finite arrays of regularly spaced elements. Our approach is based on coupling an array decomposition technique with a multipole expansion for interacting distant elements. This hybrid technique results in Toeplitz storage for both near-zone matrices and far-zone translation operators, with FFT acceleration for the far-zone element interactions. The matrix storage is of the same order as a single array element, regardless of array size, hence removing the matrix storage bottleneck for large arrays. The total storage requirements of this method are only $O(N)$, where N is the length of the solution vector. Hence, fast and rigorous analysis of very large finite arrays can be accomplished with limited resources.

1. Introduction

The Array Decomposition-Fast Multipole Method (**AD-FMM**) is an innovative approach to significantly reducing storage and CPU requirements for finite array analysis. This approach builds on the recently introduced Array Decomposition Method (**ADM**) [1]. In ADM, all interactions between array elements are carried out in the near-zone via dense integral equation methods. However, the regular spacing of the array elements results in a Toeplitz matrix system, allowing for reduced storage techniques. Similar to ADM, in AD-FMM, each element of the array is treated as an identical unit cell corresponding to a unique and regular grid point of the array lattice. However, in the hybrid approach presented here, each element undergoes a multipole expansion, and the Fast Multipole Method (FMM) [2] is then used to interact distant elements. The chosen clustering grid for FMM conforms to the array lattice, hence allowing the creation of a single unit cluster (assuming all array elements are geometrically alike). Moreover, because of the regular clustering grid, the far-zone interactions of the unit clusters are reduced to a convolution process that can be accelerated with the fast Fourier transform (FFT). Further, since the majority of the array element interactions are carried out in the far-zone with the FMM expansion, the near-zone interactions are reduced to a few 'nearest neighbor' element interactions. Combined with the Toeplitz storage of the near-zone coupling matrices, this hybrid approach results in matrix storage that is only slightly higher than that of a single element. For large systems the storage is then $O(N)$ for a unique solution vector of length N. In contract to ADM, which uses array decomposition and FFT acceleration on the near-zone terms, AD-FMM combines an array decomposition on both the near- and far-zone terms, with the FFT acceleration on the far-zone terms (near- zone interactions are highly minimized).

0-7803-7846-6/03/$17.00

2. Underlying Formulation and Decomposition Method

Though the method presented in this paper can be generalized to any integral equation formulation, we choose to apply the technique to the Finite Element-Boundary Integral (FE-BI) method, as it allows us the freedom to treat inhomogeneous materials and arbitrary element geometries [3, 4]. We treat each individual array element as a closed volume, modeling the inside of the volume with FEM elements and discretizing the boundary with surface elements, treated via boundary integral equations (BI). Without loss of generalization, it is assumed that the array elements are modeled as physically separated, with no FEM interaction between elements (element surfaces do not touch).

In a conventional FE-BI approach, the matrix is constructed and grouped based on operator type [3, 4]. However, this expansion has no inherent symmetry and preconditioning a large matrix system of this type is difficult. Consequently, in a recent paper [1], an alternative expansion was suggested that is more practical for finite array problems -- one based instead on element interactions. The suggested system expansion takes the form

total AD-FMM matrix storage

$$\begin{bmatrix} [a]_{11'} & [a]_{12'} & \cdots & [a]_{1M'} \\ [a]_{21'} & [a]_{22'} & \cdots & [a]_{2M'} \\ \vdots & \vdots & \ddots & \vdots \\ [a]_{M1'} & [a]_{M2'} & \cdots & [a]_{MM'} \end{bmatrix} \begin{Bmatrix} \{x\}_1 \\ \{x\}_2 \\ \vdots \\ \{x\}_M \end{Bmatrix} = \begin{Bmatrix} \{b\}_1 \\ \{b\}_2 \\ \vdots \\ \{b\}_M \end{Bmatrix} . \quad (1)$$

In this, the diagonal sub-matrices $[a_{mm'}]$ are complete FE-BI systems modeling the self-coupling of the individual array elements, whereas the off-diagonal sub-matrices consist solely of integral operators which couple the array elements. This near-zone expansion has a Toeplitz form, requiring unique storage of only the first row and column of sub-matrices (light gray + unshaded sub-matrices). However, once we apply the multipole expansion, the bulk of the element interactions are carried out in the far-zone, reducing the matrix storage to roughly the unshaded region in the upper right corner of the matrix. This unshaded region corresponds to typically one or two cross-coupling terms per array dimension (one or two near-neighboring elements on each side). For large systems with N unknowns, the solution and system excitation vector storage then become the dominant storage bottleneck ($O(N)$ total storage).

3. Results

To give the reader a demonstration of the accuracy and effectiveness of AD-FMM, we compare with several other established approaches for analyzing finite arrays. The arrays are constructed from wideband tapered-slot antennas as shown in Figure 1. The element is a double-sided tapered-slot antenna fed via stripline using a double-Y balun feed, with dimensions of $11.45 \times 5.0 \times 0.1524\,\text{cm}$, and modeled with 805 FEM, 892 BI unknowns at 2.4GHz. In Figure 2, it can be seen that the results for AD-FMM agree with those from the established methods. For the comparisons, of particular interest to us is the required matrix storage for each problem, the matrix fill time, the number of required iterations, as well as the iterative and complete solution time for each method. These results are summarized in Table 1. Unlike all other methods compared, it can be seen that AD-FMM has fixed matrix storage for all finite arrays analyzed, resulting in far lower storage and fill time. Further, AD-FMM uses the same matrix layout and preconditioning as ADM, and thus it can be seen that these two methods result in the same low number of iterations. As suggested in previous studies [1, 5], the matrix layout and block-diagonal preconditioning associated with array decomposition results in superior convergence properties. For AD-FMM, there is more overhead during the iterative solution process, resulting in slightly higher solution times over ADM. The solution times for conventional FE-BI or the MLFMM approach to the finite array problem do not come close to the same low solution times. Nonetheless, it is obvious that AD-FMM can be used to accurately analyze systems much larger than any of the other methods, though solution storage remains a limiting factor. In fact, with AD-FMM, it is possible to analyze a 7 million unknown array problem on a desktop computer using only main system memory. Comparisons with measurements for a similar array structure will be presented, where AD-FMM is used to analyze multiple array interactions [6].

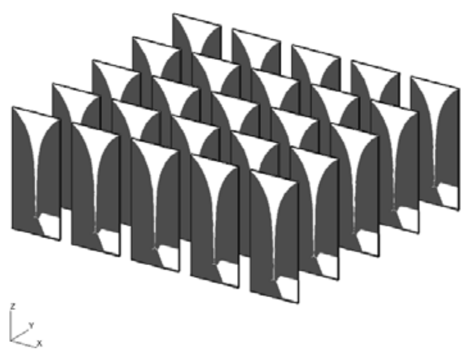
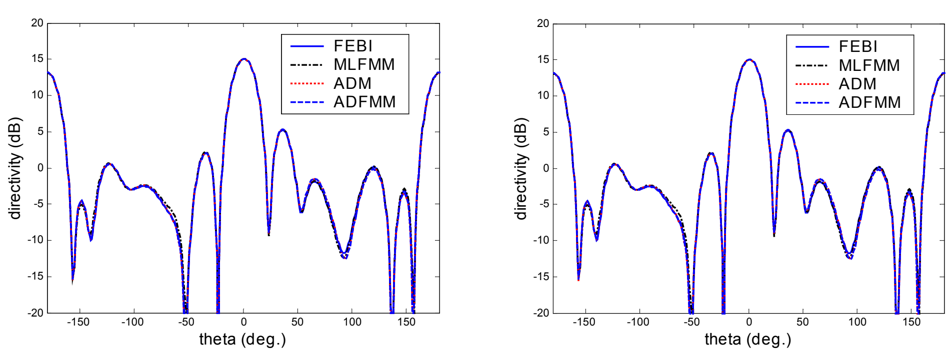


Figure 1. Element and array type used in comparisons.



(a) E-plane (b) H-plane

Figure 2. Pattern comparisons of the tested methods.

Table 1. Cost comparison for various methods and finite arrays.

Array Size		3×3	5×5	16×16	32×32	64×64
Unknowns		15,273	42,425	434,432	1,737,728	6,950,912
Matrix Storage requirements	FE-BI	1.1GB	8.5GB	868TB	13PB	217PB
	MLFMM	162MB	502MB	5.3GB	21GB	88GB
	ADM	193MB	612MB	7.2GB	29GB	121GB
	AD-FMM	193MB	193MB	193MB	193MB	193MB
Matrix fill-time	FE-BI	24m	3h	*14d	*218d	*9.5y
	MLFMM	5m	18m	6h	*28h	*5d
	ADM	8m	25m	5h	*20h	*3d
	AD-FMM	7m	7m	7m	7m	7m
Iterations	FE-BI	6	108	-	-	-
	MLFMM	6	69	-	-	-
	ADM	2	4	19	*62	*100
	AD-FMM	2	4	19	62	100
Iterative Solution Time	FE-BI	2m	1h	-	-	-
	MLFMM	1m	51m	-	-	-
	ADM	6s	27s	42m	*7h	*25h
	AD-FMM	10s	1m	1h	17h	2d
Total Solution Time	FE-BI	26m	4h	-	-	-
	MLFMM	6m	1h	-	-	-
	ADM	9m	26m	6h	*1d	*4d
	AD-FMM	8m	9m	1h	17h	2d
Total Storage Requirements	FE-BI	1.1GB	8.5GB	868TB	13PB	217PB
	MLFMM	175MB	536MB	*5.6GB	*23GB	*94GB
	ADM	194MB	613MB	7.2GB	*29GB	*121GB
	AD-FMM	200MB	201MB	214MB	258MB	432MB

*projected results

References

[1] R. Kindt, K. Sertel, E. Topsakal, and J. L. Volakis, "Array Decomposition Method for the Accurate Analysis of Finite Arrays," *to appear in IEEE Trans. Antennas Propagat.,*May 2003.

[2] R. Coifman, V. Rocklin, and S. Wandzura, "The Fast Multipole Method for the Wave Equation: A Pedestrian Prescription," in *IEEE Antennas Propagat. Mag.*, vol. 35, 1993, pp. 7-12.

[3] J. L. Volakis, A. Chatterjee, and L. C. Kempel, *Finite Element Method for Electromagnetics*. New York: IEEE Press, 1998.

[4] X. Q. Sheng, J. M. Jin, J. M. Song, C. C. Lu, and W. C. Chew, "On the Formulation of Hybrid Finite-Element and Boundary-Integral Methods for 3D Scattering," *IEEE Trans. Antennas Propagat.*, vol. 46, no. 3, pp. 303-311, March 1998.

[5] R. Kindt, K. Sertel, E. Topsakal, and J. L. Volakis, "A Domain Decomposition of the Finite Element-Boundary Integral Method for Finite Array Analysis," Applied Computational Electromagnetics Society Annual Review, pp. 103-109, Monterey, CA, 2002.

[6] R. Kindt and J. L. Volakis, "A Multi-Cell Array Decomposition Approach to Composite Finite Array Analysis," IEEE AP-S International Symposium, Columbus, OH, 2003.