7

NUMERICAL IMPLEMENTATIONS OF THE CONJUGATE GRADIENT METHOD AND THE CG-FFT FOR ELECTROMAGNETIC SCATTERING

A. F. Peterson, S. L. Ray, C. H. Chan, and R. Mittra

7.1 Introduction

There has been considerable research directed toward the development of iterative techniques for frequency-domain electromagnetic (EM) radiation and scattering problems. Since EM applications give rise to complex-valued non-Hermitian matrix equations, the conjugate gradient (CG) method appears to be superior to alternative iterative approaches. This chapter presents an overview of the numerical implementation process. In addition to a review of one CG algorithm, a primary goal of the chapter is to identify situations applicable to iterative solution methods. Fundamentally, any iterative method will only be advantageous if it can produce a solution more efficiently than LU factorization or some other direct method. Unfortunately, for general systems of equations LU factorization usually requires fewer arithmetic operations than the CG algorithm. The CG algorithm has been primarily employed to exploit structure or sparsity in the matrix equation under consideration, in order to reduce the associated memory significantly below that required for a direct method of solution. By reducing the required amount of directly-addressable memory, the CG algorithm can permit the analysis of electrically larger geometries. However, not all problems can be formulated to permit storage reduction. The formulation of EM problems will be reviewed, with emphasis placed on the creation of structure in the system matrix.

In addition to the use of the CG method to exploit matrix structure, the algorithm can be efficient if its rate of convergence is relatively fast. The convergence rate of the CG algorithm is directly related to the number of distinct eigenvalues arising in the iteration matrix and their weighting in the eigenvector decomposition of the initial residual [Peterson, Smith and Mittra, 1988]. This analysis can be used to explain the observed behavior of the CG algorithm. It is unfortunate that in many cases the algorithm does not converge at a sufficiently fast rate for the CG approach to be superior to direct methods of solution. However, direct methods may fail (due to a buildup of round-off errors) if the system matrix becomes very ill-conditioned. The CG method is sometimes proposed as an alternative algorithm for treating ill-conditioned systems, since, in principle, rounding errors do not build from one iteration step to the next. Special cases where the CG method is likely to be useful for treating poorly conditioned systems are identified.

Although certain situations may favor the use of iteration, the

trade-off between iterative and direct methods of solution is also dependent on the machine in use. Recommendations on the general efficiency of the CG method must take into account the specific computer (the word length, the amount of available fast memory, and the presence or absence of specialized architectures such as pipeline processors or highly parallel configurations). With this in mind, the information presented in this chapter is intended to aid the reader in deciding when to use the CG algorithm for a given problem.

The CG method appears to have been first employed for EM applications by Daniel and Mittra [Daniel and Mittra, 1970], who used the algorithm to treat an overdetermined system representing a two-dimensional EM scattering problem. Although the algorithm performed well, it did not receive widespread use until the early 1980s [Sarkar, Siarkiewicz and Stratton, 1981; Sarkar and Rao, 1984; van den Berg, 1984; Peterson and Mittra, 1985]. The roots of this research began in the late 1960s when Bojarski and others investigated the iterative treatment of systems containing slightly perturbed Toeplitz symmetries [Ko and Mittra, 1976; Tsao and Mittra, 1982; Kastner and Mittra, 1983abc; Borup and Gandhi, 1984; Nyo, Adams, Harrington, 1985]. These systems have the advantage of discrete-convolutional symmetries that allow the use of the FFT algorithm to implement the matrix-vector multiplications required within an iterative algorithm. The primary difficulty with these early iterative approaches was that they employed a Jacobi-type algorithm that often diverged when applied to the complex-valued non-Hermitian systems. The CG algorithm provided a convergent replacement for the simple Jacobi algorithm, and in recent years has achieved widespread use for treating systems with perturbed Toeplitz symmetries under the name "CG-FFT" [Borup and Gandhi, 1985; Sarkar, Arvas and Rao, 1985; Su, 1987; Peters and Volakis, 1988].

Numerous CG implementations have been proposed. In many of these, the algorithm is cast into a continuous form and applied directly to a continuous operator equation [van den Berg, 1984; Mittra and Chan, 1985]. This approach sometimes leads to a different way of looking at the required operations, and might suggest improvements in the way of organizing these operations. In order to implement the CG method numerically, however, the operator must be discretized into a finite-dimensional matrix [Ray and Peterson, 1988]. Thus, any numerical implementation of the CG method is equivalent to using the

CG algorithm to solve the associated matrix equation. Without loss of generality, we consider only the matrix form of the algorithm below.

7.2 Numerical Formulation of Electromagnetics Problems

The numerical treatment of a frequency-domain electromagnetics problem requires that the problem be posed in terms of a continuous equation and subsequently discretized into matrix form. Several different types of equations are employed. Open-region scattering problems can be formulated in terms of integral equations containing a singular Green's function or in terms of differential equations explicitly incorporating some form of radiation condition [Peterson, 1988]. The integral equation formulation permits the computational domain to be limited to the surface or volume of the scatterer under consideration, whereas the differential equation formulation usually requires additional unknowns to be assigned to some part of the region outside the scatterer. Although integral equation formulations generally require fewer unknowns, they discretize to produce fully- populated matrices. Differential equation formulations produce sparse matrices. In general, the matrix operators produced by either type of formulation are complex-valued and non-Hermitian.

The continuous equation to be discretized can be expressed in general operator form as

$$Lf = g \tag{7.1}$$

where L denotes the operator, f is the unknown function to be determined, and g is the given excitation (usually in the form of a field produced by some external source). This general form could represent a surface integral equation (SIE), a volume integral equation (VIE), or a volume differential equation (VDE). While several different procedures are in use to discretize these equations into matrix form, we consider only one, the method of moments (MOM) [Harrington, 1982].

The MOM discretization process begins with the selection of a set of expansion or basis functions to represent the unknown function f (the domain space of the operator L). In addition, a set of weighting or testing functions must be selected to represent the range space of

L. If f is replaced by

$$f \cong \sum_{n=1}^{N} x^{(n)} B^{(n)} \tag{7.2}$$

in (7.1), and the equation is enforced approximately by taking an inner product with each of the testing functions, the result is the discrete system

$$Ax = b \tag{7.3}$$

where x is a column vector containing the coefficients $\{x^{(n)}\}$ from (7.2), the entries of the matrix A are

$$A^{(m,n)} = \langle T^{(m)}, LB^{(n)} \rangle \tag{7.4}$$

and the column vector b contains entries

$$b^{(m)} = \langle T^{(m)}, g \rangle \tag{7.5}$$

Usually, the number of basis and testing functions is identical and A is a square $N \times N$ matrix. However, this does not have to be the case. Note that other discretization procedures, such as the finite difference method, produce similar matrix equations and in many cases can be placed on a one-to-one correspondence with a particular MOM discretization.

In Section 7.5, we will return to the question of whether or not we want to consider an iterative solution to (7.3). To a large degree, this will depend on the presence of any special symmetries or sparsity in the matrix A, which in turn depend on the nature of the operator L and the expansion and testing functions. In practice, the CG algorithm is often employed to treat equations where A contains perturbed Toeplitz symmetries. Section 7.6 describes the manner in which these symmetries can be exploited to reduce the required computer storage and computation within a CG solution. Sections 7.7–7.10 present examples illustrating perturbed Toeplitz symmetries and the CG-FFT implementation.

7.3 The Conjugate Gradient Algorithm

There are several different ways in which we can develop the CG algorithm. For example, we can construct the algorithm from the minimization of an error functional or from an orthogonal expansion of the solution. In actuality, these two ideas are linked together, i.e., each functional is associated with a specific orthogonal expansion. The CG algorithm reduces to the process of generating the orthogonal functions and finding the proper coefficients to represent the desired solution.

It suffices to consider the non-singular matrix equation

$$Ax = b \tag{7.6}$$

where A denotes an $N \times N$ matrix, x is the unknown $N \times 1$ column vector to be determined, and b is a given $N \times 1$ column vector usually denoted as the "right-hand side." It is necessary to define an inner product, and we employ the conventional Euclidean scalar product

$$\langle x, y \rangle = y^* x \tag{7.7}$$

and its associated norm

$$\|x\| = \sqrt{\langle x, x \rangle} \tag{7.8}$$

where the " $*$ " denotes transpose-conjugate matrix.

All iterative algorithms for the solution of (7.6) seek an estimate of the solution in the form

$$x_n = x_{n-1} + \alpha_n p_n \tag{7.9}$$

where x_{n-1} is a previous estimate of the solution, p_n is a "direction" vector (p_n determines the direction in the N-dimensional space in which the algorithm moves to correct the estimate of x), and α_n is a scalar coefficient (α_n determines how far the algorithm moves in the p_n direction). Although all iterative methods are similar in that they follow the form of (7.9), they differ in the procedure by which they generate α_n and p_n. Non-divergence can be guaranteed by selecting α_n in order to minimize an error functional. The CG algorithm to be presented is based on the error functional

$$E_n(x_n) = \|Ax - b\|^2 \tag{7.10}$$

Note that other functionals have been used and give rise to related members of the family of CG algorithms. The coefficient α_n from (7.9) that minimizes the functional is given by

$$\alpha_n = \frac{-\langle r_{n-1}, Ap_n \rangle}{\|Ap_n\|^2} \tag{7.11}$$

where for convenience we define the residual vector

$$r_n = Ax_n - b \tag{7.12}$$

The error functional from (7.10) is related to an orthogonality property. In order to connect these ideas, consider a solution estimate of the form

$$x_n = x_{n-1} + \alpha_n(p_n + \beta_n q_n) \tag{7.13}$$

where the direction vectors p_n and q_n are fixed, and the scalar coefficients α_n and β_n are to be obtained in order to simultaneously minimize the error functional of (7.10). Carrying out the simultaneous minimization, we find that α_n is given by (7.11) with p_n replaced by $(p_n + \beta_n q_n)$ and β_n is given by

$$\beta_n = \frac{\langle r_{n-1}, Aq_n \rangle \|Ap_n\|^2 - \langle r_{n-1}, Ap_n \rangle \langle Ap_n, Aq_n \rangle}{\langle r_{n-1}, Ap_n \rangle \|Aq_n\|^2 - \langle r_{n-1}, Aq_n \rangle \langle Aq_n, Ap_n \rangle} \tag{7.14}$$

Suppose that p_n and q_n are arbitrary direction vectors, but that q_n has been previously used in the iterative procedure, so that

$$x_{n-1} = x_{n-2} + \alpha_{n-1} q_n \tag{7.15}$$

where α_{n-1} was previously found to minimize the error functional, i.e.,

$$\alpha_{n-1} = \frac{-\langle r_{n-2}, Aq_n \rangle}{\|Aq_n\|^2} \tag{7.16}$$

It immediately follows that

$$\langle r_{n-1}, Aq_n \rangle = \langle r_{n-2} + \alpha_{n-1} Aq_n, Aq_n \rangle = 0 \tag{7.17}$$

$$\beta_n = \frac{-\langle Ap_n, Aq_n \rangle}{\|Aq_n\|^2} \tag{7.18}$$

and

$$\langle Aq_n, A(p_n + \beta_n q_n)\rangle = 0 \tag{7.19}$$

Therefore, the process of selecting direction vectors and coefficients to minimize the error functional of (7.10) is optimized when vectors satisfying the orthogonality condition

$$\langle Ap_i, Ap_j\rangle = 0 \quad i \neq j \tag{7.20}$$

are used. If an arbitrary set of direction vectors are employed, the process of minimizing (7.10) will adjust their coefficients in order to generate a sequence satisfying (7.20). Vectors which satisfy (7.20) are said to be mutually conjugate with respect to the operator A^*A, where A^* is the adjoint with respect to the inner product, i.e,

$$\langle A^*x, y\rangle = \langle x, Ay\rangle \tag{7.21}$$

In accordance with our definition for the inner product, the matrix A^* is the transpose-conjugate of A.

Suppose that a set of direction vectors satisfying the orthogonality condition of (7.20) is readily available. Since A is non-singular, these vectors are linearly independent and span the N dimensional space. The solution can be expressed in the form

$$x = x_0 + \alpha_1 p_1 + \alpha_2 p_2 + \cdots + \alpha_N p_N \tag{7.22}$$

where, for generality, the arbitrary vector x_0 can be thought of as an initial estimate or "guess" for the solution x. Because of the orthogonality of (7.20), each coefficient can be found independently according to

$$\alpha_n = \frac{-\langle r_0, Ap_n\rangle}{\|Ap_n\|^2} \tag{7.23}$$

where r_0 is defined in (7.12).

From the above relationships, it is apparent that

$$r_n = r_0 + \alpha_1 Ap_1 + \cdots + \alpha_n Ap_n \tag{7.24}$$

and recursive relationships are given as

$$r_n = r_{n-1} + \alpha_n Ap_n \tag{7.25}$$

$$x_n = x_{n-1} + \alpha_n p_n \tag{7.26}$$

and

$$\|r_n\|^2 = \|r_{n-1}\|^2 - |\alpha_n|^2 \|Ap_n\|^2 \tag{7.27}$$

From (7.20), (7.23), and (7.24) we can readily deduce that

$$\langle r_n, Ap_m \rangle = \begin{cases} \langle r_0, Ap_m \rangle & n < m \\ 0 & n \geq m \end{cases} \tag{7.28}$$

Therefore, (7.11) and (7.23) are equivalent.

In order to judge the accuracy of x_n, it is desirable to estimate the error vector

$$e_n = x - x_n \tag{7.29}$$

at each iteration step. However, this quantity is not directly computable since we do not know the solution x. Instead, it is convenient to compute the residual norm

$$N_n = \frac{\|r_n\|}{\|b\|} = \frac{\|Ax_n - b\|}{\|b\|} \tag{7.30}$$

The residual norm only provides an indirect bound on the error, according to

$$\frac{\|e_n\|}{\|e_0\|} \leq \kappa(A) \frac{\|r_n\|}{\|r_0\|} \tag{7.31}$$

where $\kappa(A)$ is the condition number of the matrix A [Golub and Van Loan, 1983]. $\kappa(A)$ is always greater than unity. If the matrix A becomes ill-conditioned, $\kappa(A)$ will grow large and the residual norm N_n may be a poor indication of the accuracy of x_n. As illustrated by (7.27), the residual norm must decrease monotonically (a direct consequence of minimizing the error norm of (7.10) at each iteration step). The CG algorithm can be terminated when the residual norm decreases to some predetermined value. As long as A is fairly well-conditioned, $N_n < 10^{-4}$ suggests that several decimal places of accuracy are obtained in the n-th iteration estimate x_n.

The above process of expanding a solution in terms of mutually conjugate direction vectors is known as the "conjugate direction method," after Hestenes and Stiefel [Hestenes and Stiefel, 1952]. The conjugate direction method does not specify the means for generating a mutually conjugate sequence, however. An approach based on the conjugate direction method which includes a procedure for generating

the p-vectors was introduced by Hestenes and Stiefel and is known as the "conjugate gradient" method. The process begins with the choice

$$p_1 = -A^* r_0 \tag{7.32}$$

which is proportional to the gradient of the functional E_n at $x = x_0$. Subsequent functions are found from

$$p_{n+1} = -A^* r_n + \beta_n p_n \tag{7.33}$$

where the scalar coefficient β_n is chosen to ensure

$$\langle A^* A p_n, p_{n+1} \rangle = 0 \tag{7.34}$$

We will demonstrate that enforcing (7.34) is sufficient to ensure that the p-vectors form a mutually conjugate set. To illustrate, we first present several relationships involving the vectors generated within the CG algorithm.

Based upon (7.33), we write

$$\langle p_{n+1}, A^* r_m \rangle = -\langle A^* r_n, A^* r_m \rangle + \beta_n \langle p_n, A^* r_m \rangle \tag{7.35}$$

From (7.28), the first and last inner product in (7.35) vanish for $m > n$, leaving

$$\langle A^* r_n, A^* r_m \rangle = 0 \quad m \neq n \tag{7.36}$$

Equations (7.28) and (7.33) can be combined to yield

$$\langle A^* r_n, p_{n+1} \rangle = -\langle A^* r_n, A^* r_n, \rangle = -\|A^* r_n\|^2 \tag{7.37}$$

It follows from (7.11) that

$$\alpha_n = \frac{\|A^* r_{n-1}\|^2}{\|A p_n\|^2} \tag{7.38}$$

From (7.25), we have

$$A^* r_n = A^* r_{n-1} + \alpha_n A^* A p_n \tag{7.39}$$

Because of the orthogonality expressed in (7.36), an inner product between $A^* r_m$ and (7.39) leads to the result

$$\langle A^* A p_n, A^* r_m \rangle = \begin{cases} \frac{\|A^* r_m\|^2}{\alpha_n} & m = n \\ \frac{-\|A^* r_m\|^2}{\alpha_n} & m = n-1 \\ 0 & \text{otherwise} \end{cases} \tag{7.40}$$

Using (7.40) with $m = n$, we find the value of β_n from (7.33) and (7.34) to be

$$\beta_n = \frac{\|A^*r_n\|^2}{\|A^*r_{n-1}\|^2} \tag{7.41}$$

To see that the formula for β_n guarantees the proper orthogonality between vectors when (7.20) is not explicitly enforced, consider the above iterative process. During the first iteration, (7.11) and (7.25) are enforced explicitly, so that

$$\langle A^*r_1, p_1 \rangle = 0 \tag{7.42}$$

Using (7.32), this is equivalent to

$$\langle A^*r_0, A^*r_1 \rangle = 0 \tag{7.43}$$

Because of (7.43), the expression for β_1 presented in (7.41) is sufficient to ensure that

$$\langle A^*Ap_2, p_1 \rangle = 0 \tag{7.44}$$

On the second iteration, (7.25), (7.42) and (7.44) guarantee that

$$\langle A^*r_2, p_1 \rangle = -\langle A^*r_2, A^*r_0 \rangle = 0 \tag{7.45}$$

Taking an inner product of (7.33) (with $n = 1$) and A^*r_2, we find that

$$\langle A^*r_1, A^*r_2 \rangle = 0 \tag{7.46}$$

Therefore, the value of β_2 from (7.41) is sufficient to ensure that

$$\langle A^*Ap_3, p_2 \rangle = 0 \tag{7.47}$$

What remains is the validity of

$$\langle A^*Ap_3, p_1 \rangle = 0 \tag{7.48}$$

From (7.33) and (7.41), p_n can be written as

$$p_n = -\|A^*r_{n-1}\|^2 \sum_{i=0}^{n-1} \frac{A^*r_i}{\|A^*r_i\|^2} \tag{7.49}$$

Using (7.49) with $n = 3$ it follows that

$$\langle A^* A p_1, p_3 \rangle = -\|A^* r_2\|^2 \sum_{i=0}^{2} \frac{\langle A^* A p_1, A^* r_i \rangle}{\|A^* r_i\|^2} \tag{7.50}$$

But, by the relationship established in (7.40) (which is valid for these values of n and m as established in (7.43), (7.45), and (7.46), the above reduces to

$$\langle A^* A p_1, p_3 \rangle = -\|A^* r_2\|^2 \left\{ \frac{-1}{\alpha_1} + \frac{1}{\alpha_1} \right\} = 0 \tag{7.51}$$

Thus, in an inductive fashion we see that the direction vectors generated by the above procedure satisfy the assumed orthogonality properties of the conjugate direction method.

In the computer science literature, this particular form of the conjugate gradient algorithm is sometimes referred to as the "conjugate gradient method applied to the normal equations." The original CG algorithm due to Hestenes and Stiefel was only valid for a Hermitian positive definite matrix A. To apply the algorithm to arbitrary systems, we have in effect premultiplied the matrix equation by A^*, constructing the "normal equations." Note that a variety of related conjugate gradient algorithms are possible, based on different error functionals or different definitions of the inner product.

The conjugate gradient algorithm is summarized as follows:

Initial steps:

$$\text{Guess } x_0$$

$$r_0 = A x_0 - b$$

$$p_1 = -A^* r_0$$

Iterate $(n = 1, 2, \ldots)$:

$$\alpha_n = -\frac{\langle r_{n-1}, A p_n \rangle}{\|A p_n\|^2} = \frac{\|A^* r_{n-1}\|^2}{\|A p_n\|^2}$$

$$x_n = x_{n-1} + \alpha_n p_n$$

$$r_n = A x_n - b = r_{n-1} + \alpha_n A p_n$$

$$\beta_n = \frac{\|A^* r_n\|^2}{\|A^* r_{n-1}\|^2}$$

$$p_{n+1} = -A^* r_n + \beta_n p_n$$

7.4 Convergence of the Conjugate Gradient Algorithm

For an arbitrary non-singular matrix A, the CG algorithm outlined in the previous section produces a solution in at most N iteration steps (assuming infinite precision arithmetic). This is a direct consequence of the fact that N p-vectors span the solution space. In addition to this desirable feature of the CG method, the solution estimates generated at each iteration step have the property that

$$\|x - x_n\| \leq \|x - x_m\| \quad n > m \tag{7.52}$$

Thus, the error in the solution estimate decreases monotonically as the algorithm progresses. To show this result, consider (7.49) and the orthogonality of the A^*r_n vectors as shown in (7.36). These equations can be combined to yield the inequality

$$\langle p_i, p_j \rangle \geq 0 \tag{7.53}$$

From the definition of x_n, it follows that

$$x_n - x_m = \sum_{i=m+1}^{n} \alpha_i p_i \tag{7.54}$$

Note that the coefficients α_i are nonnegative by (7.38). Equations (7.53) and (7.54) can be combined to produce

$$\langle x_n - x_m, x_N - x_n \rangle \geq 0 \quad N > n > m \tag{7.55}$$

Finally, (7.55) can be combined with the identity

$$\|x - x_m\|^2 = \|x_n - x_m\|^2 + \|x - x_n\|^2 + 2\mathrm{Re}\{\langle x_n - x_m, x - x_n \rangle\} \tag{7.56}$$

to prove (7.52).

Although the above analysis shows that the CG algorithm produces estimates x_n that converge monotonically to the solution of the matrix equation, it says nothing about the rate of convergence. Furthermore, it might appear that the entire set of direction vectors $\{p_1, p_2, \cdots p_N\}$ are required to produce the solution. In order to study the convergence of the CG method from a different perspective, note that the residual at the n-th iteration step can be written as

$$r_n = R_n(AA^*)r_0 \tag{7.57}$$

where $R_n(AA^*)$ is the residual polynomial of order n in the matrix AA^*, i.e.,

$$R_n(AA^*) = \sum_{k=0}^{n} \xi_{nk}(AA^*)^k \tag{7.58}$$

The coefficients $\{\xi_{nk}\}$ in (7.58) are combinations of the previous scalars α and β from (7.38) and (7.41), and are therefore real-valued [Stiefel, 1958; Jennings, 1977].

To relate the residual polynomial to the error norm produced by the CG algorithm at the n-th iteration step, let $\{\lambda_i\}$ denote the eigenvalues of the matrix AA^* (these are also the eigenvalues of the matrix A^*A) and let $\{u_i\}$ denote the orthonormal eigenvectors of AA^*. The initial residual vector can be decomposed into an eigenvector expansion according to

$$r_0 = \sum_{i=1}^{N} \langle u_i, r_0 \rangle u_i \tag{7.59}$$

Inserting this expansion into (7.57), replacing $(AA^*)^k u_i$ by $(\lambda_i)^k u_i$, and using the orthonormal property of the eigenvectors produces the result

$$E_n = \langle r_n, r_n \rangle = \sum_{i=1}^{N} |\langle u_i, r_0 \rangle|^2 [R_n(\lambda_i)]^2 \tag{7.60}$$

Equation (7.60) can be used to draw a variety of conclusions about the CG algorithm [Stiefel, 1958; Jennings, 1977; Peterson, Smith and Mittra, 1988]. First, note that at some point in the iteration process, the algorithm will terminate with $E_n = 0$. At this point, the algorithm will have generated a residual polynomial $R_n(\lambda)$ having zeros at each of the eigenvalues λ_i. Since the algorithm can place one additional zero in this polynomial at each iteration step, and will place these zeros in order to minimize E_n, it follows that the algorithm will require at most M steps to converge, where M is the number of independent eigenvalues of AA^*. In addition, if eigenvalues are repeated or clustered together in groups, the algorithm may only need to place one zero somewhere within the cluster in order to significantly reduce the error as measured by (7.60). Furthermore, the terms in (7.60) are weighted by the coefficients of the eigenvector decomposition of the initial residual. If some of these coefficients are very small, the algorithm

will not need to place a zero of the polynomial at the corresponding eigenvalues. In fact, if the initial residual can be represented by exactly one eigenvector, the CG algorithm will converge in exactly one iteration!

The effect of the initial estimate of the solution on the convergence behavior is to alter the initial residual. The choice of $x_0 = 0$ as an initial estimate of the solution produces an initial residual norm of $N_0 = 1$ in accordance with (7.30). A "good" initial guess will be one that reduces N_0 from unity to some smaller value ($N_0 = 10^{-2}$?) by altering the coefficients in (7.60) without introducing additional terms into the summation. In other words, it would be counterproductive to employ an initial estimate of the solution that excites more eigenvectors in the decomposition of (7.59) than a zero estimate for x_0 would excite, because more iteration steps would be required despite a smaller initial residual. Because of the difficulty of ensuring this property, the zero estimate for x_0 is often employed in practice and is sometimes called an "optimal" starting value. (The "best" starting value would be the solution x!)

A similar analysis could be employed to evaluate preconditioning strategies used with the CG algorithm [Evans, 1983; Kas and Yip, 1987]. Although a detailed discussion of preconditioning is beyond the scope of the present article, the technique can be viewed as the solution of the modified equation

$$M^{-1}Ax = M^{-1}b \tag{7.61}$$

where M^{-1} is an approximate inverse to the matrix A. A good preconditioning would alter the eigenvalue distribution of the iteration matrix or the eigenvector decomposition of the initial residual in order to ensure convergence in fewer steps than required by the CG algorithm applied to the original equation $Ax = b$.

7.5 When to Use the CG Method in Computational Electromagnetics

Recent research suggests that iterative methods offer advantages over direct methods in three situations [Peterson, 1987]. The most common use of iteration is to exploit some storage reduction feature present in the matrix equation. Iterative algorithms can easily exploit

any sparsity or redundancy in the matrix elements to reduce computer memory requirements, permitting the treatment of systems too large to be analyzed in other ways and improving the computational efficiency for entire classes of problems. The sparse systems arising from differential equation formulations of electromagnetics problems [Peterson, 1988] could be treated in an obvious manner using the CG method. For EM applications formulated in terms of integral equations, the CG algorithm has been widely employed to treat equations with slightly perturbed Toeplitz symmetries [Borup and Gandhi, 1985; Cwik and Mittra, 1985; Pearson, 1985; Peterson and Mittra, 1985, 1987; Peterson, 1986]. Because of its importance, this class of problems is examined in detail in Sections 7.6–7.10. A second situation in which iteration may be preferable to direct methods of solution arises if the convergence of the iterative algorithm is very rapid. As discussed in the preceding section, fast convergence is a result of the specific eigenvalue distribution and eigenvector decomposition of the excitation. A third situation where iteration may be preferable to direct methods of solution involves the failure of direct methods due to ill-conditioning. It is sometimes argued that iteration may be the only stable way to solve ill-conditioned equations [Sarkar et al., 1981]. This argument has merit, but only in special cases.

The question of whether to use the CG algorithm for a given class of problems may be brought into better focus by a discussion of when not to use the iterative algorithm. It is relatively easy to identify situations when iterative methods will be less efficient than alternative direct methods of solution. For example, if many right- hand sides are to be treated, it is doubtful that iterative methods can compete with direct methods that generate an implicit inverse matrix (i.e., in the form of an LU factorization). A similar situation arises if the convergence of the iterative method proves to be extremely slow. Because of the uncertainty associated with convergence rates, it is usually conceded that direct methods are preferable for the treatment of systems small enough to be stored in primary memory. (A modern supercomputer can store fully- populated systems having order in the thousands; specialized machines that spool data from secondary storage can treat full matrix equations having order in the tens of thousands.) Iterative methods might prove superior for specialized large-order equations that can not be stored in directly-addressable memory.

There are a variety of factors that affect the efficiency of iteration

as compared to direct methods such as LU factorization. Iteration can be terminated after a few digits of accuracy are obtained in the solution, which may be all that is desired or needed for the application at hand. If a good initial estimate of the solution is available, an iterative algorithm may be able to refine the result to necessary accuracy in far less time than required for the direct solution of the same system (which does not make use of the initial estimate). It is possible that an iterative algorithm may converge quickly even without a good initial estimate of the solution (for instance, fast convergence may result if only a few eigenvectors are excited by the initial residual). In other words, there are circumstances where iterative algorithms require less computation than direct methods. It is difficult, however, to identify these situations except by trial and error. In terms of the error functional as written in (7.60), the convergence behavior is determined by the eigenvalue distribution of the operator and the eigenvector decomposition of the initial residual. The properties of certain canonical electromagnetics problems can be studied to learn more about the typical behavior of the CG algorithm for EM applications [Peterson, Smith and Mittra, 1988]. In order to accomplish this, we need to relate the eigenvalue structure of the original continuous operator to that of the MOM matrix operator.

To relate the eigenvalues of the MOM matrix to those of the original operator, consider the continuous eigenvalue equation involving the original operator, namely,

$$LE = \lambda E \tag{7.62}$$

(Assuming the existence of solutions to this equation, E is an eigenfunction and λ an eigenvalue.) The discretization of (7.62) using the identical expansion and testing functions employed in (7.2-5) produces the generalized matrix eigenvalue equation

$$Ae = \lambda Se \tag{7.63}$$

where the entries of matrix A are defined in (7.4) and the entries of matrix S are given by

$$S_{mn} = \langle T^{(m)}, B^{(n)} \rangle \tag{7.64}$$

The inner product used in (7.64) is the same as that employed in (7.4) and (7.5). The S-matrix is a scaling matrix that alters the location of

Eigenvalues of A	*Square of mag of eigs of A*	*Eigenvalues of AA^**
$346.495 - j\ 45.1867$	122101	122101
$114.592 + j198.680$	52605.3	52605.2
$114.592 + j198.680$	52605.2	52605.2
$7.8130 + j108.948$	11930.8	11930.8
$7.8130 + j108.948$	11930.7	11930.7
$0.2265 + j\ 66.7360$	4453.74	4453.74
$0.2265 + j\ 66.7359$	4453.74	4453.73
$0.0036 + j\ 52.2904$	2734.29	2734.28
$0.0036 + j\ 52.2904$	2734.28	2734.28
$0.0000 + j\ 48.4810$	2350.41	2350.40

Table 7.1 Eigenvalues of the MOM matrix A and the matrix AA^*, for an order-10 matrix constructed from the electric-field integral equation using subsectional pulse basis functions and Dirac delta testing functions. The scatterer is a circular conducting cylinder illuminated by a TM wave. A is complex-symmetric and circulant.

the original eigenvalues in the complex plane when projecting them onto the matrix operator. Equation (7.63) suggests that the eigenvalues of the matrix $S^{-1}A$ will be some sort of approximation to the eigenvalue spectrum of the original continuous operator L. This relationship has been verified by numerical experimentation [Peterson, Smith, Mittra, 1988].

The interpretation of the CG algorithm embodied in (7.60) uses the eigenvalues of AA^* rather than those of A. For a general non-Hermitian matrix A, a precise mathematical relationship can not be found between the complex eigenvalues of A and the real eigenvalues of the Hermitian matrix AA^*. However, in special cases such as electromagnetic scattering from circular perfectly conducting cylinders, the eigenvalues of AA^* are the square of the magnitude of the eigenvalues of A (Table 7.1). Thus, there is reason to believe that some correspondence exists that can be used to link these ideas for many actual applications.

For EM scattering problems formulated in terms of integral equations and involving uniform plane wave excitations, numerical experiments have shown that approximately N/3 eigenvectors are excited significantly by the right-hand side when a discretization involving about 10 subsectional cells per wavelength is employed [Peterson and Mittra, 1986]. While this is not a general result, it is typical that far less than the full N iteration steps are required with the CG algorithm to re-

duce N_n to 10^{-4}. However, it was also observed that seldom does the CG algorithm converge in fewer than N/6 iteration steps, unless the discretization level is very fine [Peterson and Mittra, 1986]. This suggests that the CG algorithm is usually not fast enough to outperform a direct method such as LU factorization. A small body of experience suggests that the convergence rate of the CG algorithm can be quite slow when treating the sparse systems arising from a discretization of the differential equations for EM scattering [Smith, Peterson and Mittra, 1990], unless preconditioning is employed. From these results, we conclude that the CG algorithm is probably not as efficient as direct methods for solving general, fully-populated systems arising from EM problems. Only in special cases (such as described in Sections 7.6–7.10) is the CG algorithm likely to prove consistently superior.

Direct methods such as LU factorization can fail because of accumulated rounding errors in a variety of situations [Golub and Van Loan, 1983]. If the matrix is ill-conditioned, slight errors in the matrix entries can lead to gross errors in the solution regardless of the particular technique (direct or iterative) used to solve the equation. LU factorization can also fail if Wilkinson's growth factor [Golub and Van Loan, 1983] becomes relatively large during factorization, or if the matrix is very ill-conditioned. A remedy to either of these latter situations may require an increase in the precision of the calculations. If implemented in a robust manner, iterative methods are thought to be less susceptible to a progressive buildup of round-off error. However, the convergence rate of iterative algorithms depends on the condition number of the system matrix, and in practice is slow for systems that are ill-conditioned [Evans, 1983]. To understand why the convergence rate of the CG algorithm is expected to be slower for poorly conditioned systems, consider (7.60). The residual polynomial $R_n(\lambda)$ must vanish at each important eigenvalue appearing in the summation of (7.60). However, the residual polynomial has unity value at the origin ($\lambda = 0$). As the matrix becomes poorly conditioned and the eigenvalue spectrum of AA^* spreads along the positive real axis, more degrees in the polynomial are necessary in order to effectively reduce the error measured by (7.60). In addition, because of increased round-off errors during the computation of the matrix-vector operations, the finite step termination property of the CG algorithm is no longer obtained. If the equation is very badly conditioned convergence may slow to the point of stagnation [Peterson and Mittra, 1985, 1987]. Finally, because of

(7.31) the residual norm usually employed to estimate the accuracy of the solution will not be valid as A becomes poorly conditioned. Thus, a reliable criterion for terminating the CG algorithm is not apparent.

These observations cast doubt on the likely success of the CG algorithm applied to very ill-conditioned systems. However, if the initial residual is orthogonal to the eigenvectors that correspond to the near-zero eigenvalues, the matrix operator will appear to be better conditioned than it actually is. In this case, the CG algorithm should remain relatively robust and the convergence rate may be acceptable. For the purpose of discussion, we refer to eigenvalues and eigenvectors that are not excited by the initial residual as *extraneous*. Examples illustrating the successful iterative solution of poorly conditioned systems usually involve extraneous eigenvalues [Sarkar et al., 1981].

As an example, extraneous eigenvalues can arise when an extremely fine discretization is used within the numerical model of an unbounded operator. Consider the numerical treatment of the electric-field integral equation (EFIE) representing TE-wave scattering from a perfectly conducting cylinder. For circular cylinders, the spectrum of the EFIE consists of an infinite set of discrete eigenvalues which tend to be spread along the negative imaginary axis in the complex plane [Peterson, Smith and Mittra, 1988]. As the order of the matrix equation is increased, eigenvalues having progressively larger magnitudes are projected from the continuous operator to the matrix operator. Although the condition number will increase with each eigenvalue, the EM field incident upon the scatterer generally contains a decreasing contribution from each additional eigenfunction. As the order of the matrix increases, the addition of extraneous eigenvalues will degrade the condition number until the LU factorization becomes unstable because of the extraneous eigenvalues. However, iterative algorithms don't "see" the extraneous eigenvalues and may provide a stable solution.

Thus, it is possible that iteration can be used to solve certain systems that are too ill-conditioned for treatment by direct methods. However, in the absence of theoretical support indicating that near-zero eigenvalues (or very large eigenvalues) are extraneous, there is little reason to expect iteration to succeed in the solution of a very ill-conditioned system.

7.6 Discrete-Convolutional Symmetries and the CG-FFT

Electromagnetic scattering problems posed in terms of integral equation formulations require the solution of a fully-populated matrix equation. Because general purpose direct algorithms for matrix solution require the full $N \times N$ matrix to be stored in computer memory, a bottleneck is placed on the solution process for large systems. There are certain EM problems that provide a significant degree of structure in the system matrix, and these allow the bottleneck to be avoided. The class of problems to be discussed are those posed in terms of integral equations having convolutional kernels. For relatively simple geometries, these equations can sometimes be discretized to yield matrices having discrete-convolutional symmetries. The types of scattering geometries treated in this manner include flat plates and surfaces of constant curvature [Pearson, 1985; Nyo, Adams and Harrington, 1985; Peterson, 1986; Peterson and Mittra, 1987; Peters and Volakis, 1988], penetrable dielectric bodies [Borup and Gandhi, 1984, 1985; Su, 1987], and planar frequency selective surfaces [Tsao and Mittra, 1982; Cwik and Mittra, 1985; Montgomery and Davey, 1985]. Unfortunately, arbitrarily-shaped structures that are convenient to analyze with surface integral equations (such as bent wires, airplanes, etc.) do not fall into the class that naturally produce discrete-convolutional symmetries in the associated system matrix. This appears to fundamentally limit the occurrence of this type of symmetry in electromagnetics equations. Although specialized Toeplitz algorithms [Pries, 1972; Golub and Van Loan, 1983] are sometimes appropriate for the systems of interest, iterative methods offer the possibility of treating more general matrix equations. For example, the slightly perturbed Toeplitz systems discussed below can not usually be treated with conventional Toeplitz routines.

The type of matrix structure of interest contains one or more discrete convolution operations. A general discrete convolution is an operation of the form [Brigham, 1974; Oppenheim and Schafer, 1975]

$$e_m = \sum_{n=0}^{N-1} j_n \, g_{m-n} \tag{7.65}$$

where e, j, and g denote sequences of numbers. Equation (7.65) is

equivalent to the matrix equation

$$\begin{bmatrix} g_0 & g_{-1} & g_{-2} & \cdots & g_{1-N} \\ g_1 & g_0 & g_{-1} & \cdots & g_{2-N} \\ g_2 & g_1 & g_0 & \cdots & g_{3-N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_{N-1} & g_{N-2} & g_{N-3} & \cdots & g_0 \end{bmatrix} \begin{bmatrix} j_0 \\ j_1 \\ \vdots \\ j_{N-1} \end{bmatrix} = \begin{bmatrix} e_0 \\ e_1 \\ \vdots \\ e_{N-1} \end{bmatrix} \tag{7.66}$$

The $N \times N$ matrix depicted in (7.66) is a general *Toeplitz* matrix. All of the elements of this matrix are described by the $2N-1$ entries of the first row and column. If the elements of the sequence g repeat with period N, so that

$$g_{n-N} = g_n \qquad n = 1, 2, \ldots, N-1 \tag{7.67}$$

the operation is known as a *circular* discrete convolution (and the $N \times N$ matrix in (7.66) is *circulant*). Otherwise, the operation is a *linear* discrete convolution. Note that any linear discrete convolution of length N can be embedded into a circular discrete convolution of length $2N-1$. This can be accomplished by extending the original sequence g to repeat with period $2N-1$, zero-padding the sequence j to length $2N-1$, and changing the upper limit of the summation in (7.65) to $2N-2$.

The fast Fourier transform (FFT) algorithm is an efficient way of implementing the discrete Fourier transform [Brigham, 1974]

$$\tilde{g}_n = \sum_{k=0}^{N-1} g_k \, e^{-j\frac{2\pi nk}{N}} \qquad n = 0, 1, \ldots, N-1 \tag{7.68}$$

The inverse discrete Fourier transform is defined

$$g_k = \frac{1}{N} \sum_{n=0}^{N-1} \tilde{g}_n \, e^{j\frac{2\pi nk}{N}} \qquad k = 0, 1, \cdots, N-1 \tag{7.69}$$

For notational purposes, we use

$$\tilde{g} = \mathrm{FFT}_N(g) \tag{7.70}$$

$$g = \mathrm{FFT}_N^{-1}(\tilde{g}) \tag{7.71}$$

to denote the discrete Fourier transform pair for a sequence of length N. The discrete convolution theorem states that if (7.65) is a circular discrete convolution of length N, it is equivalent to

$$\tilde{e}_n = \tilde{j}_n \tilde{g}_n \qquad n = 0, 1, \cdots, N-1 \tag{7.72}$$

If Equation (7.65) is a linear discrete convolution, the equivalence holds if the linear convolution is embedded in a circular convolution of length $2N-1$ as described above.

To summarize, the discrete convolution operation of (7.65) is equivalent to the Toeplitz matrix multiplication of (7.66). Furthermore, either can be implemented using the FFT and inverse FFT algorithm according to the discrete convolution theorem [Brigham, 1974; Oppenheim and Schafer, 1975]

$$e = \mathrm{FFT}_N^{-1}\{\mathrm{FFT}_N(j)\mathrm{FFT}_N(g)\} \tag{7.73}$$

If the discrete convolution is of the linear type, the FFT's must be of length $2N-1$ rather than length N.

The above conclusions are easily generalized to two or three dimensions. A two-dimensional discrete convolution is an operation of the form

$$e_{pq} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} j_{nm}\, g_{p-n,q-m} \qquad \begin{Bmatrix} p \\ q \end{Bmatrix} = 0, 1, \cdots, N-1 \tag{7.74}$$

This equation is equivalent to the matrix operation

$$\begin{bmatrix} G_0 & G_{-1} & \cdots & G_{1-N} \\ G_1 & G_0 & \cdots & G_{2-N} \\ \vdots & \vdots & \ddots & G_{3-N} \\ G_{N-1} & G_{N-2} & \cdots & G_0 \end{bmatrix} \begin{bmatrix} J_0 \\ J_1 \\ \vdots \\ J_{N-1} \end{bmatrix} = \begin{bmatrix} E_0 \\ E_1 \\ \vdots \\ E_{N-1} \end{bmatrix} \tag{7.75}$$

where each element of the $N \times N$ *block Toeplitz* matrix of (7.75) is itself a Toeplitz matrix of the form depicted in (7.66). The relationship established in (7.73) can be extended to multiple dimensions in an obvious manner.

To illustrate the appearance of discrete convolutional structure in electromagnetics equations, consider the one-dimensional integral equation

$$E(x) = \int_a^b J(x')K(x-x')\,dx' \qquad a < x < b \tag{7.76}$$

$J(x)$ represents the unknown function to be determined and E and K are given. If we consider an application of the method of moments as described in Section 7.2, under the restriction that basis and testing functions have the form

$$B^{(n)}(x) = B(x - x_n) \tag{7.77}$$

$$T^{(m)}(x) = T(x - x_m) \tag{7.78}$$

where

$$x_n = x_0 + n\Delta x \tag{7.79}$$

the discrete system described in (7.3) can be written as

$$e_m = \sum_{n=1}^{N} j_n g_{m-n} \tag{7.80}$$

which is exactly the discrete-convolutional form identified above. In Equation (7.80),

$$g_{m-n} = \int_{-\infty}^{\infty} T(x - x_m) \int_{-\infty}^{\infty} B(x' - x_n) K(x - x')\, dx'\, dx \tag{7.81}$$

In contrast to direct methods, iterative algorithms only require the presence of an implicit matrix operator (a subroutine that when given a column vector returns the product of the $N \times N$ system matrix with the column vector). As a result, any type of structure in the matrix can be easily exploited using iterative algorithms. The matrix structure or sparsity can be completely accounted for in the subroutine that performs the matrix-vector multiplication. Therefore, the specific type of matrix structure need not affect the organization of the part of the computer program that involves the main body of the iterative algorithm. The CG driver routine can be thought of as a "black box" similar in form to library routines that perform Gaussian elimination. Although the perturbed Toeplitz structure is probably the most common arising from integral equation formulations, it only occurs in special cases involving relatively simple geometries. In many other practical problems, most of the $N \times N$ matrix will not contain any type of structure.

To illustrate the appearance of discrete-convolutional symmetries in actual electromagnetic problems, the following sections present examples of several applications of the CG-FFT, including scattering

from inhomogeneous dielectric cylinders, hollow finite-length circular cylinders, resistive or conducting plates, and frequency selective surfaces. In each case, the problem formulation and method-of-moments discretization will be discussed, followed by an appraisal of several advantages and disadvantages of each particular CG-FFT implementation.

7.7 TM-Wave Scattering by Inhomogeneous Dielectric Cylinders

Figure 7.1 depicts an inhomogeneous dielectric cylinder that can be characterized by a complex relative permittivity $\varepsilon_r(x,y)$. If illuminated by a normally incident transverse magnetic (TM) wave, the field components present are E_z, H_x, and H_y. The dielectric material may be replaced by equivalent polarization currents radiating in free space, defined

$$\bar{J}(x,y) = \hat{z}j\omega\varepsilon_0[\varepsilon_r(x,y) - 1]E_z(x,y) \tag{7.82}$$

Although this equivalent volumetric source is an unknown function, it must satisfy the electric field integral equation (EFIE)

$$E_z^{inc}(x,y) = \frac{J_z}{j\omega\varepsilon_0(\varepsilon_r - 1)} + j\omega\mu_0 A_z \tag{7.83}$$

where

$$A_z(x,y) = \iint J_z(x',y')\frac{1}{4j}H_0^{(2)}(kR)dx'dy' \tag{7.84}$$

$$R = \sqrt{(x-x')^2 + (y-y')^2} \tag{7.85}$$

and E_z^{inc} denotes the known incident field (the field that would be present in the absence of the dielectric material).

The cylinder cross-section can be divided into cells as illustrated in Fig. 7.2. Each cell in the model is assumed to have a constant average relative permittivity ε_{rn} that may vary from cell to cell. However, the cell size and shape is constrained to follow the lattice structure depicted in Fig. 7.2, in order to provide the necessary symmetry features needed to employ the CG-FFT. For the moment, assume that the cells in the model are numbered in some random fashion from 1 to N. If the

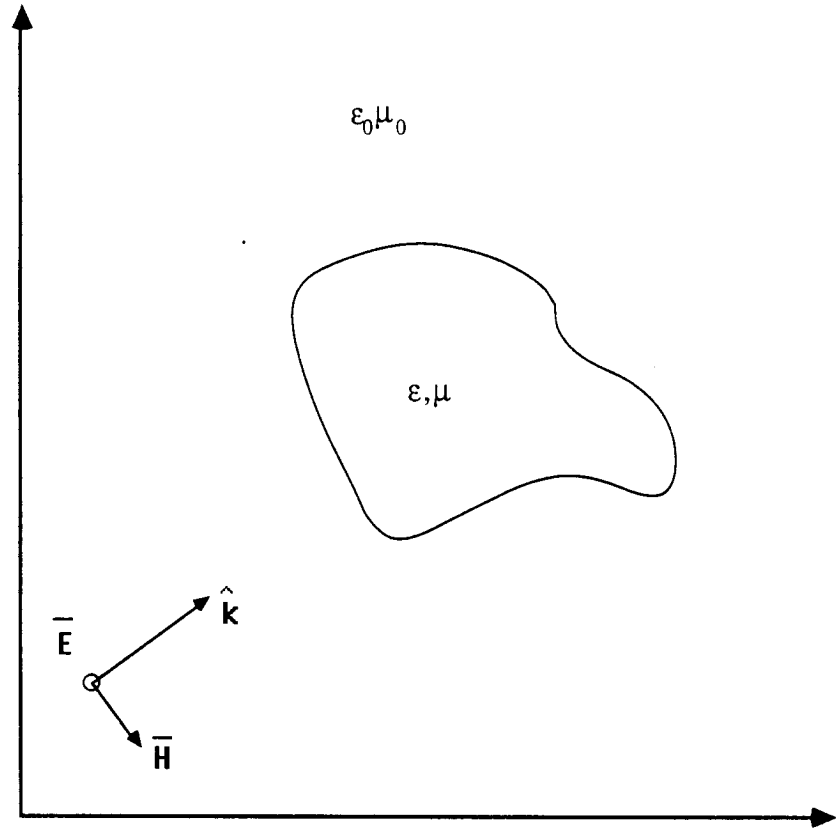

Figure 7.1 Cross-sectional view of a dielectric cylinder illuminated by a TM wave.

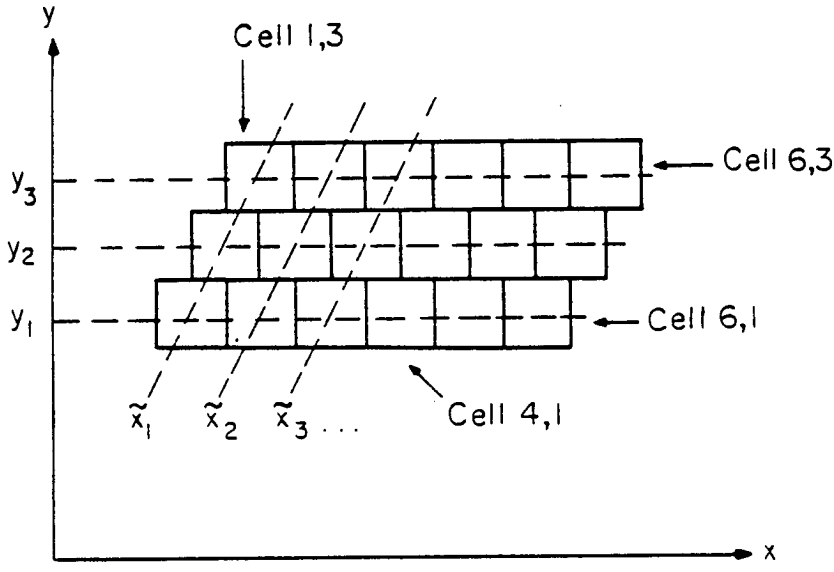

Figure 7.2 Regular lattice of square cells used to model the cylinder cross section.

unknown polarization current density is represented within each cell by a constant or "pulse" basis function

$$p_n(x,y) = \begin{cases} 1 & \text{if } (x,y) \in \text{cell } n \\ 0 & \text{otherwise} \end{cases} \tag{7.86}$$

so that the global representation for the current density is the superposition

$$J_z(x,y) \cong \sum_{n=1}^{N} j_n p_n(x,y) \tag{7.87}$$

where j_n are scalar coefficients to be determined, (7.83) reduces to

$$E_z^{inc}(x,y) \cong \sum_{n=1}^{N} j_n \left\{ \frac{\eta p_n(x,y)}{jk[\varepsilon_r(x,y)-1]} + jk\eta \int\int_{\text{cell } n} \frac{1}{4j} H_0^{(2)}(kR)dx'dy' \right\} \tag{7.88}$$

Enforcing (7.88) at the centers of each of the N cells produces an $N \times N$ system

$$\begin{bmatrix} E_z^{inc}(x_1,y_1) \\ E_z^{inc}(x_2,y_2) \\ \vdots \\ E_z^{inc}(x_N,y_N) \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & \cdots & Z_{1N} \\ Z_{21} & Z_{22} & \cdots & Z_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{N1} & Z_{N2} & \cdots & Z_{NN} \end{bmatrix} \begin{bmatrix} j_1 \\ j_2 \\ \vdots \\ j_N \end{bmatrix} \tag{7.89}$$

whose entries are given by

$$Z_{mn} = \frac{k\eta}{4} \iint_{\text{cell } n} H_0^{(2)}(kR_m)dx'dy' \qquad m \neq n \tag{7.90}$$

and

$$Z_{mm} = \frac{\eta}{jk(\varepsilon_{rm}-1)} + \frac{k\eta}{4} \iint_{\text{cell } m} H_0^{(2)}(kR_m)dx'dy' \tag{7.91}$$

where

$$R_m = \sqrt{(x_m - x')^2 + (y_m - y')^2} \tag{7.92}$$

The integrals in (7.90) and (7.91) can be evaluated in closed-form if the cell shapes are approximated by circles of the same area [Richmond, 1965]. The necessary integration is given by

$$\int_{\phi'=0}^{2\pi} \int_{\rho'=0}^{a} H_0^{(2)}(kR)\rho' d\rho' d\phi' = \begin{cases} \frac{2\pi a}{k} J_0(k\rho) H_1^{(2)}(ka) - \frac{j4}{k^2} & \rho < a \\ \frac{2\pi a}{k} J_1(ka) H_0^{(2)}(k\rho) & \rho > a \end{cases} \tag{7.93}$$

where (ρ, ϕ) represent conventional cylindrical coordinates. Using the circular-cell approximation, we obtain

$$Z_{mn} = \frac{\eta \pi a_n}{2} J_1(ka_n) H_0^{(2)}(kR_{mn}) \qquad m \neq n \tag{7.94}$$

where a_n is the equivalent radius of cell n and

$$R_{mn} = \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2} \tag{7.95}$$

The relative permittivity ε_r only appears in the diagonal matrix entries

$$Z_{mm} = \frac{\eta \pi a_m}{2} H_1^{(2)}(k a_m) - \frac{j \eta \varepsilon_{rm}}{k(\varepsilon_{rm} - 1)} \tag{7.96}$$

We now consider the lattice numbering suggested in Fig. 7.2. Assuming that the lattice consists of N_x by N_y cells numbered globally by rows parallel to the y-axis, the numbering follows the organization $1, 2, \ldots, N_y, N_y+1, \ldots, 2N_y, \ldots N_x N_y$. Because of the lattice structure, the matrix elements of (7.94) are only functions of the displacement along the lattice relative to the source cell. It follows that the Z-matrix from (7.89) can be written in the form of an N_x by N_x block Toeplitz matrix

$$\begin{bmatrix} Z_0^{(0)} & Z_1 & Z_2 & \cdots & Z_{Nx-1} \\ Z_1 & Z_0^{(1)} & Z_1 & \cdots & Z_{Nx-2} \\ Z_2 & Z_1 & Z_0^{(2)} & \cdots & Z_{Nx-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Z_{Nx-1} & Z_{Nx-2} & Z_{Nx-3} & \cdots & Z_0^{(Nx-1)} \end{bmatrix} \tag{7.97}$$

where each block has the N_y by N_y Toeplitz form

$$\begin{bmatrix} z_0 & z_1 & z_2 & \cdots & z_{Ny-1} \\ z_1 & z_0 & z_1 & \cdots & z_{Ny-2} \\ z_2 & z_1 & z_0 & \cdots & z_{Ny-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ z_{Ny-1} & z_{Ny-2} & z_{Ny-3} & \cdots & z_0 \end{bmatrix} \tag{7.98}$$

The entries located along the main diagonal of (7.97) may be perturbed from the Toeplitz structure because of the presence of ε_{rm} in (7.96).

The Toeplitz/block Toeplitz structure of (7.89) can be alternatively expressed in terms of a two-dimensional linear discrete convolution

$$E_{mn}^{inc} = D_{mn}J_{mn} + \sum_{p=0}^{Ny-1}\sum_{q=0}^{Nx-1} J_{pq}K_{m-p,n-q}$$
$$m = 0,1,\ldots,N_y - 1$$
$$n = 0,1,\ldots,N_x - 1 \qquad (7.99)$$

where we have now adopted a two-dimensional indexing scheme to denote the relative location in the lattice of Fig. 7.2. E_{mn}^{inc} is the incident field sampled at location mn in the lattice,

$$D_{mn} = \frac{-j\eta\varepsilon_{rmn}}{k(\varepsilon_{rmn} - 1)} \qquad (7.100)$$

$$K_{0,0} = \frac{\eta\pi a}{2}H_1^{(2)}(ka) \qquad (7.101)$$

$$K_{i,j} = \frac{\eta\pi a}{2}J_1(ka)H_0^{(2)}(k\tilde{R}_{ij}) \qquad (7.102)$$

and $\tilde{R}_{ij}$ denotes the distance between the centers of two cells displaced by i cells along y and j cells along $\tilde{x}$ in the lattice of Fig. 7.2.

The redundancy present in the matrix is clearly indicated by the Toeplitz structure of (7.97) and (7.98), and in fact the entire Z-matrix can be generated from at most $2N_xN_y$ entries. Thus, the equation is an excellent candidate for iterative solution. Iterative solution algorithms do not explicitly require the presence of the $N \times N$ matrix, since they only use the result of the matrix multiplied with a column vector. Because the matrix does not need to be stored in computer memory, the necessary storage can be reduced from $(N_xN_y)^2$ to $O(N_xN_y)$. Since the matrix operator is primarily a linear discrete convolution, it can be implemented using a two-dimensional FFT algorithm. Although the required matrix-vector multiplications could be implemented explicitly, the FFT permits a reduction in computation from $O(N^2)$ to $O(N \log N)$ and adds to the overall efficiency of the implementation. However, there is a price to pay for the use of the FFT in this manner. Because (7.99) involves a linear rather than a circular convolution, it is necessary to employ zero-padding with the FFT. In this two-dimensional problem, zero-padding effectively quadruples the required storage associated with the arrays used to store the sequences

J and K appearing in (7.99). (Although this is a relatively small price to pay in this example, in a three-dimensional vector problem the storage overhead could grow by more than a factor of 24 because of zero padding.)

Note that to ensure the accuracy of the numerical approach the cells must be small in terms of the wavelength in the dielectric medium, defined as

$$\lambda_d = \frac{1}{\sqrt{|\varepsilon_r|}}\lambda_0 \tag{7.103}$$

As a "rule of thumb," a minimum of 100 cells per square dielectric wavelength is recommended. Although the formulation is capable of treating a highly inhomogeneous cylinder, the constraint of the equal-cell-size lattice geometry means that the cell size is dictated by the wavelength in the cell having greatest relative permittivity. This may impose a smaller cell size than necessary on much of the scatterer.

In general, the lattice shape may not coincide with the cross-sectional shape of the cylinder under consideration. One possibility is to assign any cell outside of the actual scatterer relative permittivity $\varepsilon_{rmn} = 1$, and treat it as a part of the cylinder. However, this unnecessarily introduces additional unknowns into the system of equations. A more efficient scheme is to employ "dummy cells" to fill out the lattice. The idea is to incorporate terms for every location in the lattice into the sequences J and K used within the discrete convolution. After each convolution is computed via the FFT and inverse FFT, locations that correspond to dummy cells in the resulting array are set equal to zero. In this manner, the iterative algorithm does not see additional unknowns at these cells and may converge faster.

The accuracy of the overall numerical formulation has been demonstrated in the literature [Richmond, 1965]. CG-FFT implementations of this formulation were developed independently by several research groups in the early 1980s; we refer the reader to the literature for additional implementation details and numerical results [van den Berg, 1984; Borup and Gandhi, 1985; Su 1987]. The convergence rate of the conjugate gradient method is relatively rapid for this formulation [Peterson and Mittra, 1986].

7.8 Scattering from Finite-Length, Hollow Conducting Right-Circular Cylinders

The scattering of electromagnetic waves by a finite-length, hollow, perfectly conducting or resistive circular cylinder can also be formulated in a manner enabling the use of the CG-FFT. This example, which includes the hollow linear dipole antenna as a special case, will also illustrate the combination of the so-called body-of-revolution (BOR) formulation with the CG-FFT. Initially, we consider perfectly conducting material, which may be replaced by equivalent electric currents radiating in free space. In accordance with the coordinate system of Fig. 7.3, there are $\hat{z}$ and $\hat{\phi}$ components of the current density present. A suitable form of the electric field integral equation is given by

$$\hat{n} \times \bar{E}^{inc} = -\hat{n} \times (-j\omega\mu_0 \overline{A} - \nabla\Phi_e) \tag{7.104}$$

where $\bar{E}^{inc}$ denotes the known incident electromagnetic field and $\hat{n}$ is the outward normal vector to the surface of the cylinder. The magnetic vector and electric scalar potentials are defined

$$\overline{A}(a,\phi,z) = \int_{z'=z_0}^{z_N} \int_{\phi'=0}^{2\pi} \bar{J}(\phi',z') \frac{e^{-jkR}}{4\pi R} a d\phi' dz' \tag{7.105}$$

$$\Phi_e(a,\phi,z) = \frac{-1}{j\omega\varepsilon_0} \int_{z'=z_0}^{z_N} \int_{\phi'=0}^{2\pi} (\nabla' \cdot \bar{J}) \frac{e^{-jkR}}{4\pi R} a d\phi' dz' \tag{7.106}$$

where

$$R = \sqrt{(z-z')^2 + 4a^2 \sin^2\left(\frac{\phi-\phi'}{2}\right)} \tag{7.107}$$

Note that (7.104) is valid only on the surface of the original cylinder.

For this cylindrical geometry, the unknown current densities, the incident fields, and the Green's function are periodic in ϕ. Each of these quantities can be expressed as a Fourier series according to

$$J_\phi(\phi',z') = \sum_{m=-\infty}^{\infty} J_{\phi m}(z') e^{jm\phi'} \tag{7.108}$$

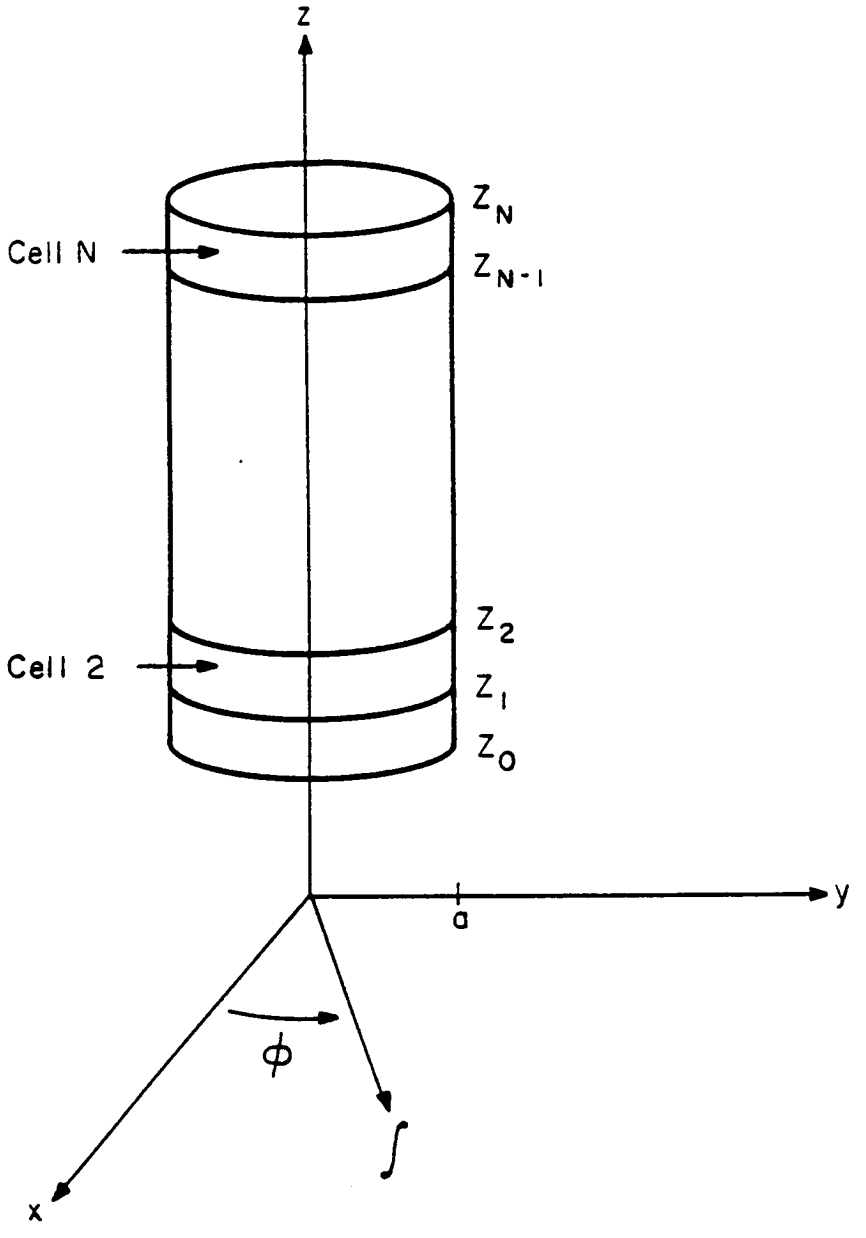

Figure 7.3 Geometry of the hollow cylinder under consideration.

$$J_z(\phi', z') = \sum_{m=-\infty}^{\infty} J_{zm}(z') e^{jm\phi'} \tag{7.109}$$

$$E_\phi^{inc}(\phi, z) = \sum_{m=-\infty}^{\infty} E_{\phi m}^{inc}(z) e^{jm\phi} \tag{7.110}$$

$$E_z^{inc}(\phi, z) = \sum_{m=-\infty}^{\infty} E_{zm}^{inc}(z) e^{jm\phi} \tag{7.111}$$

and

$$\frac{e^{-jkR}}{4\pi R} = \sum_{m=-\infty}^{\infty} G_m(z - z') e^{jm(\phi - \phi')} \tag{7.112}$$

where $J_{\phi m(z')}$ and $J_{zm}(z')$ are unknowns to be determined,

$$E_{\phi m}^{inc}(z) = \frac{1}{2\pi} \int_{\alpha=-\pi}^{\pi} E_\phi^{inc}(a, \alpha, z) e^{-jm\alpha} d\alpha \tag{7.113}$$

$$E_{zm}^{inc}(z) = \frac{1}{2\pi} \int_{\alpha=-\pi}^{\pi} E_z^{inc}(a, \alpha, z) e^{-jm\alpha} d\alpha \tag{7.114}$$

and

$$G_m(z - z') = \frac{1}{2\pi} \int_{\alpha=-\pi}^{\pi} \frac{e^{-jk\tilde{R}}}{4\pi\tilde{R}} e^{-jm\alpha} d\alpha \tag{7.115}$$

where

$$\tilde{R} = \sqrt{(z - z')^2 + 4a^2 \sin^2\left(\frac{\alpha}{2}\right)} \tag{7.116}$$

Substituting the above expansions into the integral equation and taking an inner product of both sides with the function

$$\frac{1}{2\pi} e^{-jp\phi} \tag{7.117}$$

separates the original equation into independent equations for each of the Fourier harmonics. The coupled system for the m-th harmonic is given by

$$-E_{\phi m}^{inc}(z) = \frac{2\pi a}{j\omega\varepsilon_0} \int_{z'=z_0}^{z_N} \left\{ k^2 J_{\phi m}(z') \frac{G_{m-1}(z - z') + G_{m+1}(z - z')}{2} + \left[-\frac{m^2}{a^2} J_{\phi m}(z') + j\frac{m}{a} \frac{\partial J_{zm}}{\partial z'} \right] G_m(z - z') \right\} dz' \tag{7.118}$$

$$-E_{zm}^{inc}(z) = \frac{2\pi a}{j\omega\varepsilon_0}\int_{z'=z_0}^{z_N} k^2 J_{zm}(z')G_m(z-z')dz' + \frac{2\pi a}{j\omega\varepsilon_0}\frac{\partial}{\partial z}\int_{z'=z_0}^{z_N}\left\{j\frac{m}{a}J_{\phi m}(z') + \frac{\partial J_{zm}}{\partial z'}\right\}G_m(z-z')dz' \tag{7.119}$$

Equation (7.115) can be simplified to the form

$$G_m(z-z') = \frac{1}{4\pi^2}\int_{\alpha=0}^{\pi}\frac{e^{-jk\tilde{R}}}{\tilde{R}}\cos(m\alpha)d\alpha \tag{7.120}$$

where $\tilde{R}$ is given in (7.116).

If the cylinder is divided into cells as illustrated in Fig. 7.3, the unknown current densities may be expanded according to

$$J_{\phi m}(z) \cong \sum_{n=1}^{N} j_{\phi n}p(z; z_{n-1}, z_n) \tag{7.121}$$

$$J_{zm}(z) \cong \sum_{n=1}^{N-1} j_{zm}t(z; z_{n-1}, z_n, z_{n+1}) \tag{7.122}$$

where $p(x; x_1, x_2)$ and $t(x; x_1, x_2, x_3)$ denote subsectional pulse and triangle basis functions, respectively, which are defined in Fig. 7.4. If these expansions are substituted into the equations for the m-th harmonic of the unknown current density, (7.118) and (7.119) may be enforced approximately via an inner product with the testing functions

$$T_\phi^p(z) = (\Delta z)_p\delta\left(z - \frac{z_{p-1}+z_p}{2}\right) \tag{7.123}$$

$$T_z^p(z) = p\left(z; \frac{z_{p-1}+z_p}{2}, \frac{z_p+z_{p+1}}{2}\right) \tag{7.124}$$

(also defined in Fig. 7.4) to produce the $(2N-1)\times(2N-1)$ matrix equation

$$\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{bmatrix}\begin{bmatrix}\mathbf{j}_\phi\\ \mathbf{j}_z\end{bmatrix} = \begin{bmatrix}E_\phi^{inc}\\ E_z^{inc}\end{bmatrix} \tag{7.125}$$

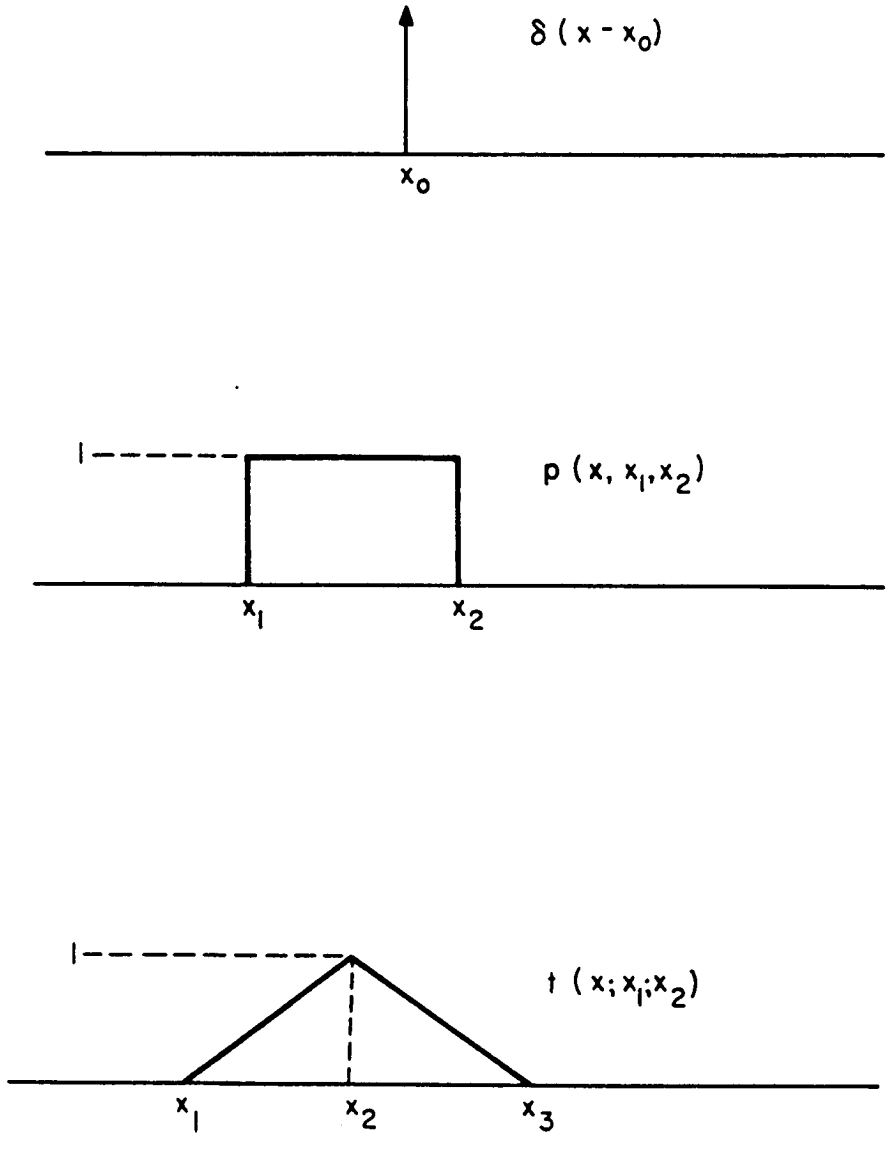

Figure 7.4 Definition of the basis and testing functions.

where

$$E_{\phi p}^{inc} = -(\Delta z)_p E_{\phi m}^{inc}\left(\frac{z_{p-1}+z_p}{2}\right) \qquad p = 1,2,\ldots,N \tag{7.126}$$

$$E_{zp}^{inc} = -\int_{\frac{z_{p-1}+z_p}{2}}^{\frac{z_p+z_{p+1}}{2}} E_{zm}^{inc}(z)dz \qquad p = 1,2,\ldots,N-1 \tag{7.127}$$

$$\begin{aligned} A_{pn} = &-ja\eta(\Delta z)_p\left\{\frac{m^2}{a^2}\int_{z_{n-1}}^{z_n} G_m\left(\frac{z_{p-1}+z_p}{2}-z'\right)dz'\right. \\ &\left.-k^2\int_{z_{n-1}}^{z_n}\frac{G_{m-1}\left(\frac{z_{p-1}+z_p}{2}-z'\right)+G_{m+1}\left(\frac{z_{p-1}+z_p}{2}-z'\right)}{2}dz'\right\} \\ &n = 1,2,\ldots,N \\ &p = 1,2,\ldots,N \end{aligned} \tag{7.128}$$

$$\begin{aligned} B_{pn} = &(\Delta z)_p m\eta\left\{\frac{1}{(\Delta z)_n}\int_{z_{n-1}}^{z_n} G_m\left(\frac{z_{p-1}+z_p}{2}-z'\right)dz'\right. \\ &\left.-\frac{1}{(\Delta z)_{n+1}}\int_{z_n}^{z_{n+1}} G_m\left(\frac{z_{p-1}+z_p}{2}-z'\right)dz'\right\} \\ &n = 1,2,\ldots,N-1 \\ &p = 1,2,\ldots,N \end{aligned} \tag{7.129}$$

$$\begin{aligned} C_{pn} = &m\eta\left\{\int_{z_{n-1}}^{z_n} G_m\left(\frac{z_p+z_{p+1}}{2}-z'\right)dz'-\right. \\ &\left.\int_{z_{n-1}}^{z_n} G_m\left(\frac{z_{p-1}+z_p}{2}-z'\right)dz'\right\} \\ &n = 1,2,\ldots,N \\ &p = 1,2,\ldots,N-1 \end{aligned} \tag{7.130}$$

$$\begin{aligned} D_{pn} = -ja\eta\left\{k^2\int_{\frac{z_{p-1}+z_p}{2}}^{\frac{z_p+z_{p+1}}{2}}\int_{z_{n-1}}^{z_{n+1}} t(z';z_{n-1},z_n,z_{n+1})G_m(z-z')dz'dz\right. \\ \left.+\frac{1}{(\Delta z)_n}\int_{z_{n-1}}^{z_n} G_m\left(\frac{z_p+z_{p+1}}{2}-z'\right)\right. \end{aligned}$$

$$
\begin{aligned}
&- G_m\left(\frac{z_{p-1}+z_p}{2}-z'\right)dz' \\
&+\frac{1}{(\Delta z)_{n+1}}\int_{z_n}^{z_{n+1}} G_m\left(\frac{z_{p-1}+z_p}{2}-z'\right) \\
&-G_m\left(\frac{z_p+z_{p+1}}{2}-z'\Big)\right)dz'\Big\} \\
&p=1,2,\ldots,N-1 \\
&n=1,2,\ldots,N-1
\end{aligned}
\tag{7.131}
$$

and where G_m is defined in (7.120).

If the cells in the model are constrained so each has the same length Δz, the above expressions simplify considerably to yield

$$
A_{pn} = -ja\eta\left\{k^2(\Delta z)K^m_{p-n} - \frac{m^2}{a^2}(\Delta z)I^m_{p-n}\right\} \tag{7.132}
$$

$$
B_{pn} = m\eta\left\{I^m_{p-n} - I^m_{p-n-1}\right\} \tag{7.133}
$$

$$
C_{pn} = m\eta\left\{I^m_{p-n+1} - I^m_{p-n}\right\} \tag{7.134}
$$

$$
\begin{aligned}
D_{pn} \cong -ja\eta\Big\{&k^2(\Delta z)I^m_{p-n} + \frac{1}{(\Delta z)} \\
&\left[I^m_{p-n+1} - 2I^m_{p-n} + I^m_{p-n-1}\right]\Big\}
\end{aligned}
\tag{7.135}
$$

where

$$
I^m_q = \int_{-\frac{\Delta z}{2}}^{\frac{\Delta z}{2}} G_m(q\Delta z - z')dz' \tag{7.136}
$$

and

$$
K^m_q = \frac{I^{m-1}_q + I^{m+1}_q}{2} \tag{7.137}
$$

Under this restriction, the sub-matrices of the system in (7.125) have a considerable amount of structure due to the manner in which the

equation was discretized. A and D are symmetric Toeplitz matrices of order N and $N-1$ respectively. The B-matrix is not square, but has the Toeplitz structure

$$B = \begin{bmatrix} -b_1 & -b_2 & -b_3 & \cdots & -b_{N-1} \\ b_1 & -b_1 & -b_2 & \cdots & -b_{N-2} \\ b_2 & b_1 & -b_1 & \cdots & -b_{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{N-2} & b_{N-3} & b_{N-4} & \cdots & -b_1 \\ b_{N-1} & b_{N-2} & b_{N-3} & \cdots & b_1 \end{bmatrix} \tag{7.138}$$

The C-matrix is related to the B-matrix by

$$C = -B^T \tag{7.139}$$

Because of the Toeplitz symmetries, all of the elements of the above matrices can be generated from the first rows of the A, B, and D systems. This amounts to a considerable degree of redundancy which can be exploited to reduce the necessary storage requirements for large values of N, if an iterative method is used to solve (7.125). In addition, since these matrix operations can be written as linear discrete convolutions, the FFT can be incorporated if desired to speed the matrix-vector multiplications as discussed in Section 7.2.

Note that (7.125) must be solved for all of the significant Fourier harmonics excited by the incident field, including both positive and negative values of m. From an inspection of (7.136), we see that

$$I_q^m = I_q^{-m} = I_{-q}^m = I_{-q}^{-m} \tag{7.140}$$

and

$$K_q^m = K_q^{-m} = K_{-q}^m = K_{-q}^{-m} \tag{7.141}$$

Thus, the A and D submatrices are independent of the sign of m, and the B and C submatrices change sign with m. As a consequence, it is not necessary to recompute the matrix entries in order to treat both positive and negative harmonics.

To illustrate the performance of the CG-FFT method described in this section, Fig. 7.5 shows the magnitude of the current density produced on a 20 wavelength cylinder by an axially incident plane wave. This result required the solution of one matrix equation of order

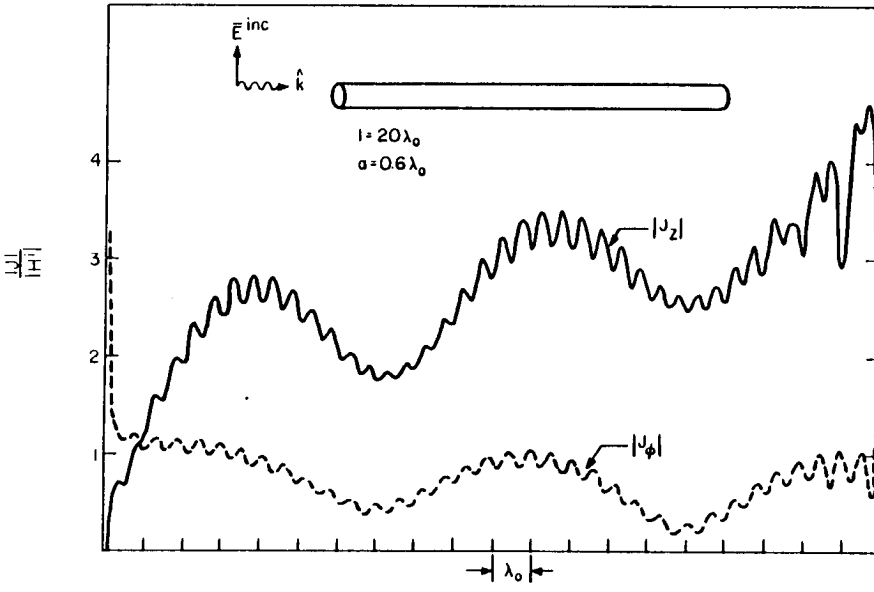

Figure 7.5 Numerical solution for the current density induced on a perfectly conducting cylinder.

399, which was accomplished by the conjugate gradient method in 79 iteration steps. The current density shows interference effects caused by superimposing interior and exterior currents, as is necessary when using the electric field integral equation to model thin structures.

The above formulation is readily extended to treat hollow cylinders constructed of resistive material, provided that the surface resistance of the resulting structure is independent of azimuthal variation. In this case, (7.104) is modified to the form

$$\hat{n} \times \bar{E}^{inc} = \hat{n} \times R_s \bar{J} - \hat{n} \times (-j\omega\mu_0 \overline{A} - \nabla\Phi) \tag{7.142}$$

where R_s is the surface resistance. The presence of the additional term in (7.142) may cause a perturbation of the Toeplitz structure along the main diagonal and immediately adjacent diagonals of the A and D matrices, but will not affect the remaining discrete convolutional structure present in the matrix equation.

Because of the restriction on the geometry needed to impose the body-of-revolution formulation and the discrete-convolutional struc-

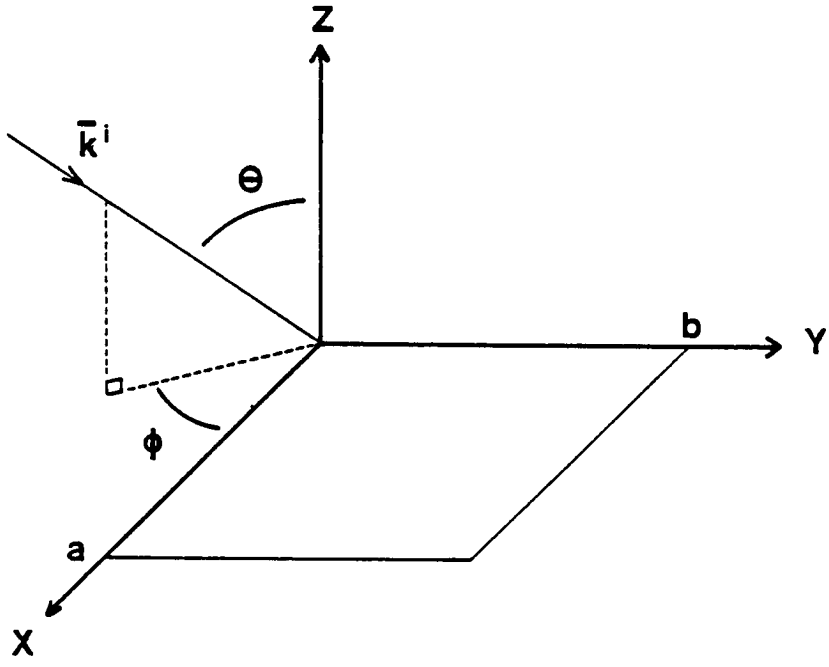

Figure 7.6 Geometry of the resistive plate.

ture, this particular example is quite specialized. As a result, few CG applications involving a body-of-revolution approach have appeared to date [Davidson and McNamara, 1988].

7.9 Scattering from Perfectly Conducting or Resistive Plates

Figure 7.6 illustrates a conducting or resistive plate located in the $z = 0$ plane and illuminated by an incident electromagnetic field. In common with the preceding formulations, the plate may be replaced by equivalent electric currents radiating in free space, which in turn can be determined from a solution of the electric field integral equation

$$\hat{n} \times \bar{E}^{inc} = \hat{n} \times R_s \bar{J} + \hat{n} \times (j\omega\mu_0 \overline{A} + \nabla\Phi) \qquad (7.143)$$

where R_s is the surface resistance. The potential functions are defined

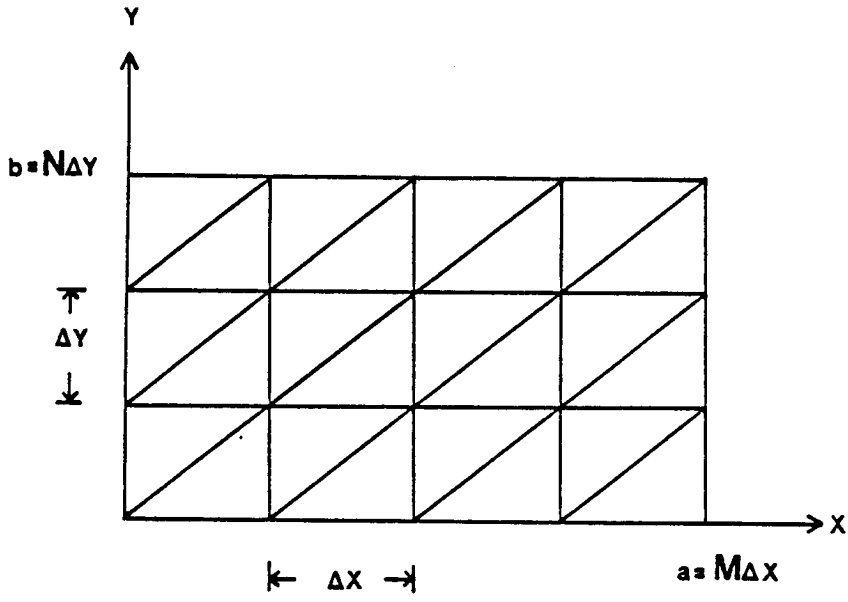

Figure 7.7 Discretization of the plate.

by

$$\overline{A}(x,y,z) = \int_{x'}\int_{y'} \bar{J}(x',y')\frac{e^{-jkR}}{4\pi R}dx'dy' \tag{7.144}$$

$$\Phi(x,y,z) = \frac{-1}{j\omega\varepsilon_0}\int_{x'}\int_{y'} (\nabla' \cdot \bar{J})\frac{e^{-jkR}}{4\pi R}dx'dy' \tag{7.145}$$

where

$$R = \sqrt{(x-x')^2 + (y-y')^2 + z^2} \tag{7.146}$$

Although (7.144) and (7.145) are written for general locations of the observer, the EFIE of (7.143) is valid only on the surface of the plate $(0 < x < a, 0 < y < b, z = 0)$.

Consider the discretization of the rectangular plate geometry into triangular patches as depicted in Fig. 7.7, so that there are M subsections of width Δx along the x coordinate and N subsections of width Δy along the y coordinate. The vertex in the lower left corner of rectangle mn is located at $(m\Delta x, n\Delta y)$. The lower and upper

triangles in rectangle mn will be denoted T_{mn}^{+} and T_{mn}^{-}. The current density may be represented by the superposition of "triangular rooftop" functions [Rao, Wilton and Glisson, 1982]

$$\bar{J}(x,y) = \sum_{i=1}^{3} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} J_{imn} \bar{f}_{imn}(x,y) \tag{7.147}$$

where $\{J_{imn}\}$ are scalar coefficients to be determined and the basis functions are defined as

$$\bar{f}_{1mn}(x,y) = \frac{L_1}{\Delta x \Delta y} \begin{cases} [x-(m-1)\Delta x]\hat{x} + [y-n\Delta y]\hat{y} & \text{in } T_{(m-1)n}^{+} \\ [(m+1)\Delta x - x]\hat{x} + [(n+1)\Delta y - y]\hat{y} & \text{in } T_{mn}^{-} \\ 0 & \text{elsewhere} \end{cases} \tag{7.148}$$

$$\bar{f}_{2mn}(x,y) = \frac{L_2}{\Delta x \Delta y} \begin{cases} [x-(m+1)\Delta x]\hat{x} + [y-n\Delta y]\hat{y} & \text{in } T_{mn}^{+} \\ [m\Delta x - x]\hat{x} + [(n+1)\Delta y - y]\hat{y} & \text{in } T_{mn}^{-} \\ 0 & \text{elsewhere} \end{cases} \tag{7.149}$$

and

$$\bar{f}_{3mn}(x,y) = \frac{L_3}{\Delta x \Delta y} \begin{cases} [x-(m+1)\Delta x]\hat{x} + [y-(n+1)\Delta y]\hat{y} & \text{in } T_{mn}^{+} \\ [m\Delta x - x]\hat{x} + [(n-1)\Delta y - y]\hat{y} & \text{in } T_{m(n-1)}^{-} \\ 0 & \text{elsewhere} \end{cases} \tag{7.150}$$

where L_i is the length of the interior edge of a type i basis function. These basis functions are shown in Fig. 7.8. The physical constraint that the current density be confined to the plate dictates that J_{10n} and J_{3m0} be set to zero.

If the expansion of (7.147) is substituted into the electric field integral equation, a Galerkin testing procedure (using testing functions identical to the basis functions) can be employed to generate the discrete system

$$E_{abc} = \sum_{imn} J_{imn} R_{abc,imn} + \sum_{imn} J_{imn} G_{b-m,c-n}^{ai} \tag{7.151}$$

where a and i assume the values $\{1,2,3\}$, b and m assume $\{0,1,\ldots,M-1\}$, and c and n assume $\{0,1,\ldots,N-1\}$ (with the

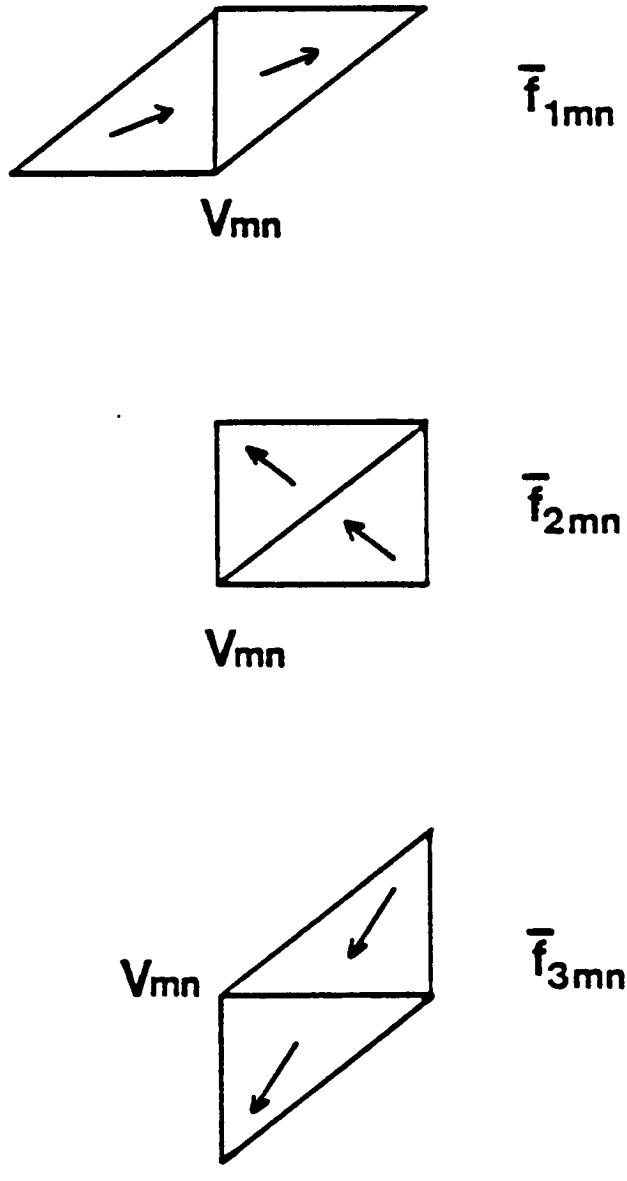

Figure 7.8 Triangular rooftop basis functions for the plate problem.

exception of certain edge terms as mentioned above). The remaining expressions are defined

$$E_{abc} = \int_x \int_y \bar{E}^{inc}(x,y) \cdot \bar{f}_{abc}(x,y) \tag{7.152}$$

$$R_{abc,imn} = \int_x \int_y R_s(x,y) \bar{f}_{imn}(x,y) \cdot \bar{f}_{abc}(x,y) \tag{7.153}$$

and

$$G^{ai}_{b-m,c-n} = \int_x \int_y (j\omega\mu_0 \overline{A} + \nabla\Phi)_{imn} \cdot \bar{f}_{abc}(x,y) \tag{7.154}$$

Equation (7.151) is a matrix equation that can be solved to determine the coefficients $\{J_{imn}\}$.

Equation (7.154) involves a quadruple integration which is approximated for reasons of numerical efficiency. The first term in (7.154) can be evaluated according to

$$\int_x \int_y \overline{A}_{imn}(x,y) \cdot \bar{f}_{abc}(x,y) \cong \frac{\Delta x \Delta y}{2} \left\{ \overline{A}_{imn}\left(\bar{r}^+_{abc}\right) \cdot \bar{f}_{abc}(\bar{r}^+_{abc}) + \overline{A}_{imn}(\bar{r}^-_{abc}) \cdot \bar{f}_{abc}(\bar{r}^-_{abc}) \right\} \tag{7.155}$$

where $\bar{r}^+_{abc}$ and $\bar{r}^-_{abc}$ are the centroids of the plus and minus triangles associated with the test function $\bar{f}_{abc}$. The second term in (7.154) is evaluated by recognizing that

$$\int_x \int_y \nabla\Phi_{imn} \cdot \bar{f}_{abc}(x,y) = -\int_x \int_y \Phi_{imn} \nabla \cdot \bar{f}_{abc}(x,y) \tag{7.156}$$

and using approximations similar to those in (7.155). The remaining double integrations required in (7.155) and (7.156) must be computed by numerical integration. There are actually only three different types of required integrals, namely

$$I^{1\pm}_{pq} = \int_0^{\Delta x} \int_0^{\frac{\Delta y}{\Delta x}x'} \frac{e^{-jkR^{\pm}_{pq}}}{R^{\pm}_{pq}} dy' dx' \tag{7.157}$$

$$I_{pq}^{2\pm} = \int_0^{\Delta x} \int_0^{\frac{\Delta y}{\Delta x}x'} x' \frac{e^{-jkR_{pq}^{\pm}}}{R_{pq}^{\pm}} dy'dx' \tag{7.158}$$

and

$$I_{pq}^{3\pm} = \int_0^{\Delta x} \int_0^{\frac{\Delta y}{\Delta x}x'} y' \frac{e^{-jkR_{pq}^{\pm}}}{R_{pq}^{\pm}} dy'dx' \tag{7.159}$$

where

$$R_{pq}^{\pm} = \left|\bar{r}_{pq}^{\pm} - \bar{r}'\right| \tag{7.160}$$

$$\bar{r}_{pq}^{\pm} = (p\Delta x + \alpha^{\pm}\Delta x)\hat{x} + (q\Delta y + \beta^{\pm}\Delta y)\hat{y} \tag{7.161}$$

$(\alpha^+, \beta^+) = (2\Delta x/3, \Delta y/3)$, the centroid of T_{00}^+, and $(\alpha^-, \beta^-) = (\Delta x/3, 2\Delta y/3)$, the centroid of T_{00}^-. There are $6(2N+1)(2M+1)$ numerical integrations required to completely specify the sequence G^{ai} appearing in (7.151).

The conjugate gradient algorithm can be applied directly to (7.151) to determine the coefficients $\{J_{imn}\}$. In fact, because of the lattice structure of the plate (Fig. 7.7), the discrete operator is convolutional in form, and the summation can be performed using two-dimensional FFTs. In total, nine FFTs are required to implement each application of the discrete operator within the iterative solution process. (Although the first term in (7.151) is not convolutional if the surface resistance is a function of location, it is readily implemented by direct multiplication since $R_{abc,imn} = 0$ whenever the basis and testing functions do not overlap.

Although we have assumed that the plate is rectangular, plates of other shape may be treated using the "dummy cell" approach described in Section 7.7. (Because of the particular triangular-cell discretization of Fig. 7.7, the plate contour may require approximation by rectangular cells in some places.)

To illustrate the performance of the approach, Fig. 7.9 shows numerical results for the surface current density induced on a square plate constructed of both perfectly conducting and resistive material. The plate has side dimension of three wavelengths, with the leading and trailing edge constructed of resistive material linearly graded from $R_s = 0$ at the perfect conductor (which occupied approximately the middle two-thirds of the plate) to $R_s = 500\Omega$ at the edges. The solution required $M = N = 32$ and 3008 unknowns.

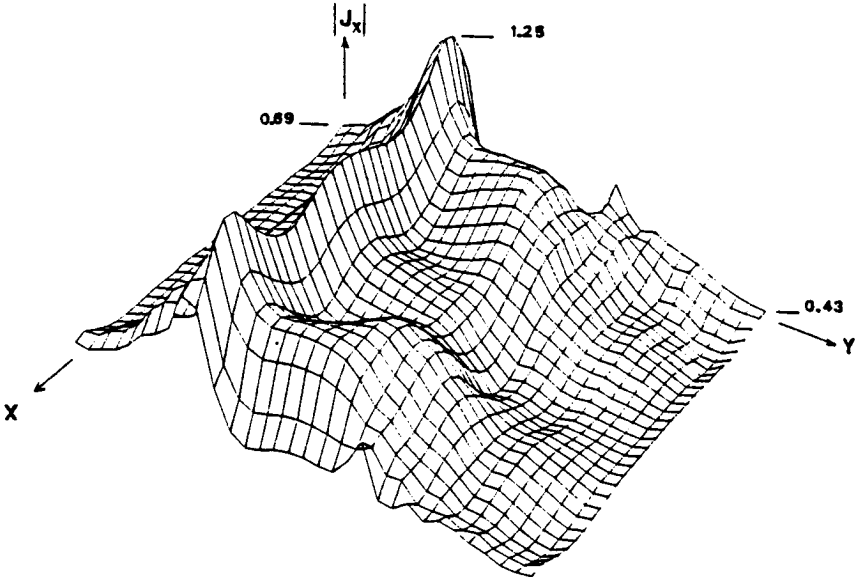

Figure 7.9 Magnitude of the x component of the induced current on a plate with resistive edge loading. Edge loading is confined to the leading 0.56λ and trailing 0.56λ of the $3\lambda \times 3\lambda$ plate, and is graded from 0Ω at the center region to 500Ω at the edges. The incident E-field is propagating from a spherical angle with $\theta = 60°$ and $\phi = -90°$ and is polarized parallel to the x-axis.

Other applications of the CG-FFT procedure to planar structures have been described in the literature [Pearson, 1985; Peters and Volakis, 1988; Catedra, Cuevas and Nuno, 1988].

Although the CG-FFT approach can permit the convenient analysis of relatively large scatterers, in general iterative methods suffer in comparison to direct methods of solution when used to treat multiple incident fields. For example, suppose it is desired to determine the solution for plane waves impinging on a square plate from 91 different incidence angles. Table 7.2 shows the execution times of both the CG-FFT approach and conventional LU factorization for perfectly conducting plates ranging in size from 1 square wavelength to 25 square

	CG				LU		
$a=b$	no. of	FFT	avg. no. of	avg. CPU	matrix	CPU (sec)	
(λ)	unknowns	size	iterations	(sec)	size	1 RHS	91 RHS
1.000	176	16x16	42	5.56	176x176	1.71	5.17
2.000	736	32x32	43	16.26	736x736	65.54	138.21
3.125	3008	64x64	58	68.50	–	–	–
5.078	12160	128x128	78	616.52	–	–	–

Table 7.2 Comparison of the CPU time (CRAY XMP/24) required to solve the system of equations for scattering from a perfectly conducting square plate of dimension $a \times b$ by the CG-FFT algorithm and conventional LU factorization with forward and back substitution. Solution times are presented for a single excitation or right-hand side (RHS) and for 91 different excitations. The averages shown are those obtained by averaging over the 91 different excitations.

wavelengths. For the 2λ by 2λ plate (736 unknowns), the CG-FFT approach produced an acceptable solution for one incident field in about 16 seconds (one fourth of the time required by LU factorization). However, if the CG-FFT is restarted anew for each of 91 incident angles, it is an order of magnitude slower than LU factorization. It follows that the tradeoff between iterative and direct methods is largely dependent on the number of incident fields under consideration. It is noteworthy, however, that the CG-FFT can solve much larger-order systems than easily treated by direct methods. For instance, Table 7.2 indicates that the CG-FFT was able to solve a system of order 12160 in 617 seconds. The storage requirements associated with LU factorization for this matrix were beyond the machine limits.

7.10 Analysis of Frequency Selective Surfaces

As a final example, we extend the previous formulation for electromagnetic scattering from a single plate to scattering from a doubly periodic infinite array of plates (i.e., a frequency selective surface). Figure 7.10 shows a portion of such a structure. For simplicity, we assume that each of the plates is identical in shape and resistivity, that the lattice defining the periodicity is aligned with the x- and y-axes, and that the incident field is a uniform plane wave of the general form

$$\bar{E}^{inc}(x,y,z) = \bar{E}_0 e^{-j(k_x^{inc}x + k_y^{inc}y + k_z^{inc}z)} \tag{7.162}$$

where

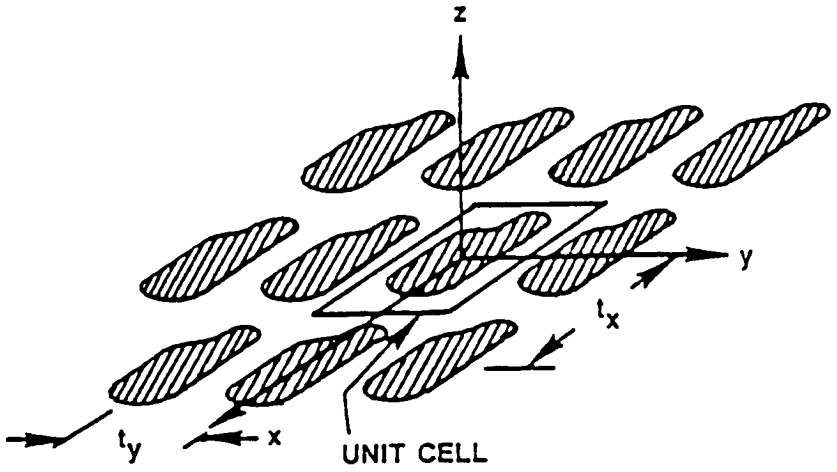

Figure 7.10 A portion of an infinite array of arbitrarily-shaped plates.

$$(k_x^{inc})^2 + (k_y^{inc})^2 + (k_z^{inc})^2 = k^2 \tag{7.163}$$

The individual plates are numbered with indices n and m denoting the location along the x-axis and y-axis, respectively. The array has period t_x in the x direction and t_y in the y direction. The current density on plate nm is related to the current density on plate 00 by a phase shift according to

$$\bar{J}(x + nt_x, y + mt_y) = \bar{J}(x, y)e^{-j(k_x^{inc} nt_x + k_y^{inc} mt_y)} \tag{7.164}$$

Thus, it suffices to consider the current on plate 00 as the primary unknown to be determined. If this plate is represented by equivalent electric currents radiating in free space, the solution can be obtained from an EFIE of the form

$$\hat{n} \times \bar{E}^{inc} = \hat{n} \times R_s \bar{J} - \hat{n} \times \frac{\nabla\nabla \cdot + k^2}{j\omega\varepsilon_0} \bar{A} \tag{7.165}$$

where

$$\overline{A}(x,y) = \iint_{\text{plate 00}} \bar{J}(x',y')G(x-x',y-y')dx'dy' \tag{7.166}$$

$$G(x-x',y-y') = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} \frac{e^{-jkR_{mn}}}{4\pi R_{mn}} e^{-j(k_x^{inc}nt_x + k_y^{inc}mt_y)} \tag{7.167}$$

and

$$R_{mn} = \sqrt{(x-x'-nt_x)^2 + (y-y'-mt_y)^2} \tag{7.168}$$

This EFIE is almost identical in form to that used in the preceding section, the difference being the periodic Green's function of (7.167). This Green's function is convolutional in x and y, however, and the EFIE can be discretized in much the same manner as the single plate formulation of Section 7.9.

Consider the discretization of plate 00 according to Fig. 7.7, and the representation of the current density on this plate in terms of the triangular rooftop functions defined in (7.147)–(7.150). If Galerkin's method is applied to convert the EFIE into a discrete system similar in form to (7.151), it is apparent that the only difference in the problem formulation is the computation of the entries $G^{ai}_{b-m,c-n}$ appearing in (7.154). The numerical evaluation of these entries is complicated by the fact that the summation of (7.167) is very slowly convergent. Acceleration procedures for evaluating these integrals have been developed, and we refer the reader to the literature for the implementation details [Chan, 1988; Mittra, Chan and Cwik, 1988]. The convolutional form present in the individual plate formulation of Section 7.9 is also present in the periodic plate formulation, and thus the CG-FFT approach can be used to solve the discrete system.

Equation (7.167) is a function of the incident field, which prescribes the phase delay seen at each cell in the infinite lattice. As a result, the entries of the sequence G depend on the angle of incidence, and must be recomputed for each incident field under consideration. Thus, the relative advantage of LU factorization over CG-FFT for treating multiple incident fields does not apply to this periodic

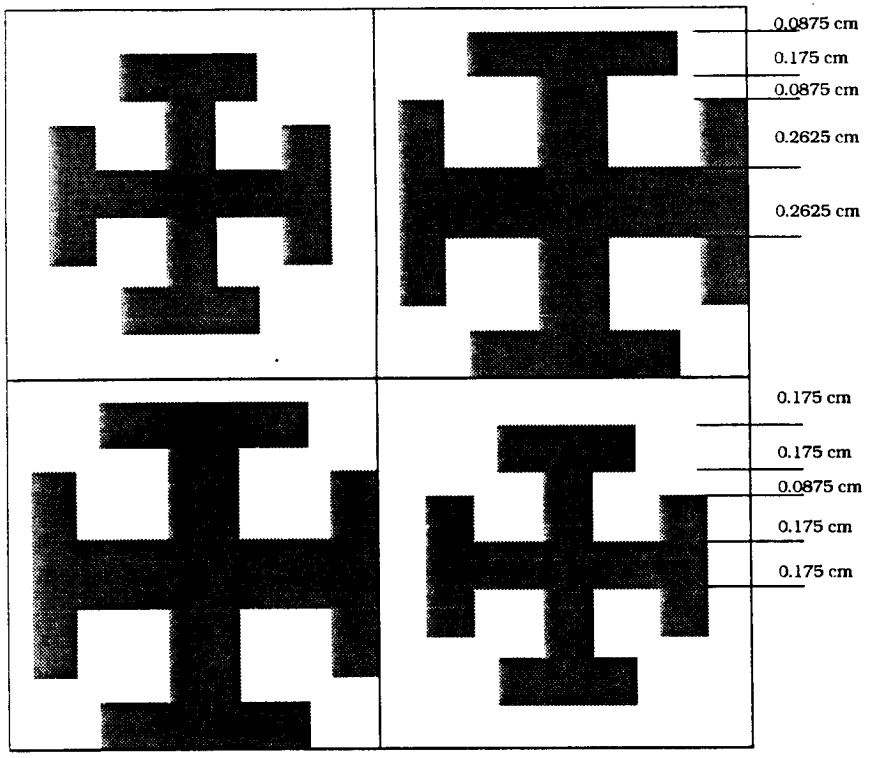

Figure 7.11 Composite unit cell of a frequency selective surface made from Jerusalem crosses of differing size.

structure in the same way it might apply to scattering from an individual plate. This may explain the widespread use of the CG-FFT for the analysis of periodic structures of this general class [Mittra, Chan and Cwik, 1988].

To illustrate the performance of the approach, Fig. 7.11 shows a unit cell associated with a particular frequency selective surface constructed of Jerusalem crosses of differing size. Together, these four crosses can be thought of as plate 00 in the above formulation. The power reflected from the corresponding infinite surface is displayed in Fig. 7.12 as a function of frequency. The particular calculation required

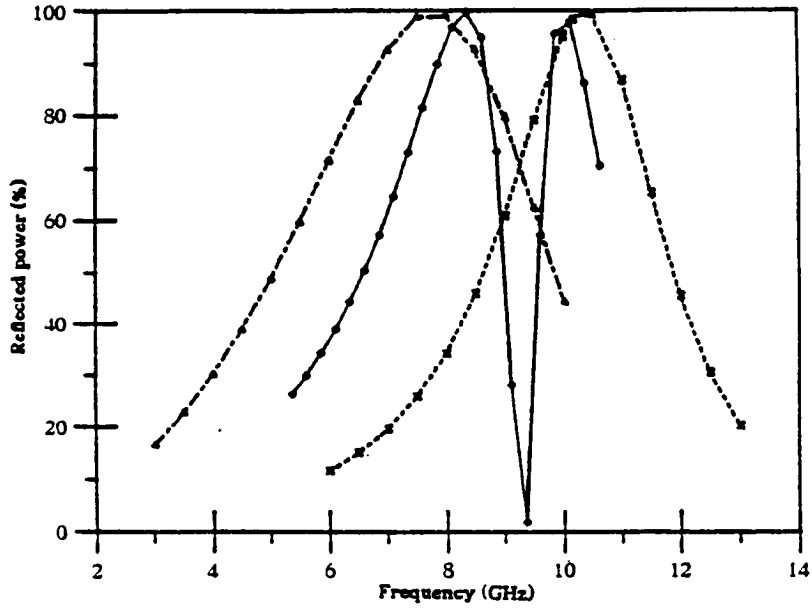

Figure 7.12 Frequency response of the periodic structure when illuminated by a normally incident plane wave.
×–×–× surface made entirely of the small Jerusalem crosses
–△–△– surface made entirely of the large Jerusalem crosses
——— surface shown in Fig. 7.11

two-dimensional FFTs of size 32×32, and a total of 616 unknowns.

7.11 The Treatment of Multiple Right-Hand Sides with the CG Algorithm

Electromagnetic scattering problems often require the repeated solution of a matrix equation in order to treat additional right-hand sides. For radar cross section applications, hundreds or thousands of different incident fields must be analyzed. Thus, there has been considerable interest in procedures to improve the relative inefficiency of iterative methods for treating multiple right-hand sides.

The most straightforward approach for the iterative treatment of multiple excitations requires the algorithm to be restarted anew for each excitation angle, incorporating an initial estimate of the solution

extrapolated from previous solutions for other angles of incidence. Provided that the solution for one angle of incidence is similar to that for another angle, this procedure can be very efficient. However, we have observed this to be effective only when the incremental angle is less than a few degrees. In addition, although this technique works well with certain geometries (i.e., flat strips), it has not been as successful with more complicated scatterers.

Since the conjugate gradient (CG) algorithms generate orthogonal direction vectors that eventually span the solution space, an alternative approach is to simultaneously expand multiple solutions in terms of a single set of the direction vectors. The bulk of the required CG computation arises from the generation of the direction vectors, and significant computational savings would result if multiple solutions were expanded simultaneously in this manner. However, several difficulties prevent this from being a trivial task. First, the direction vectors generated by the CG algorithm will be orthogonal to any matrix eigenvectors absent from the specific right-hand side used to start the CG process. While this is partially responsible for the relatively quick convergence of the CG algorithm in practice, it suggests that the direction vectors are geared to represent the solution corresponding to one specific right-hand side and will not be optimum for the representation of other solutions. In addition, round-off errors cause a progressive loss of orthogonality, and will prevent the set of direction vectors from spanning the solution space.

In spite of these difficulties, some progress has been made in using the CG algorithm for the treatment of multiple right-hand sides [Smith, Peterson and Mittra, 1989]. Features that make a multiple excitation algorithm possible include the use of a composite system to serve as an initial "seed" for the generation of the direction vectors and the systematic restarting of the algorithm using a new seed when required to maintain reasonable orthogonality between direction vectors. Although this algorithm is generally not competitive with direct methods, it appears to be an improvement over completely restarting the CG procedure for each right-hand side.

Consider the solution of the family of matrix equations $Ax^{(m)} = b^{(m)}$. Each solution is represented in terms of a single set of the direction vectors p_i, so that the estimate of the solution for the m^{th}

excitation at the n^{th} iteration is

$$x_n^{(m)} = x_0^{(m)} + \sum_{i=1}^{n} \alpha_i^{(m)} p_i \tag{7.169}$$

The corresponding m^{th} residual at the n^{th} iteration is defined $r_n^{(m)} = Ax_n^{(m)} - b^{(m)}$. The coefficients can be computed according to

$$\alpha_n^{(m)} = -\frac{\langle r_{n-1}^{(m)}, Ap_n \rangle}{\|Ap_n\|^2} \tag{7.170}$$

The required overhead to treat multiple right-hand sides includes computing $\alpha_n^{(m)}$ for each solution estimate and updating the unknowns $x_n^{(m)}$ and the residuals $r_n^{(m)}$. The complete algorithm is as follows:
Initial steps:

$$\text{Guess } x_0^{(m)}$$

$$r_0^{(m)} = Ax_0^{(m)} - b$$

$$p_1 = -A^* r_0^{\text{System used as seed}}$$

Iterate $(n = 1, 2, \ldots)$:
(*For the system used as seed compute:*)

$$\alpha_n^{(\text{System used as seed})} = \frac{\|A^* r_{n-1}^{(\text{System used as seed})}\|^2}{\|Ap_n\|^2}$$

(*All other systems use:*)

$$\alpha_n^{(m)} = -\frac{\langle r_{n-1}^{(m)}, Ap_n \rangle}{\|Ap_n\|^2}$$

(*All systems:*)

$$x_n^{(m)} = x_{n-1}^{(m)} + \alpha_n^{(m)} p_n$$

$$r_n^{(m)} = r_{n-1}^{(m)} + \alpha_n^{(m)} Ap_n$$

$$\beta_n = \frac{\|A^* r_n^{(\text{System used as seed})}\|^2}{\|A^* r_{n-1}^{(\text{System used as seed})}\|^2}$$

$$p_{n+1} = -A^* r_n^{(\text{System used as seed})} + \beta_n p_n$$

As discussed in Section 7.4, this CG algorithm will terminate before N (the order of the system) iterations if the excitation is orthogonal to one or more eigenvectors of $AA^{\star}$, where $A^{\star}$ denotes the transpose-conjugate of A. This situation poses a problem for any multiple right-hand-side algorithm, and motivates the use of a "composite" excitation as the seed for generating the direction vectors. The composite excitation is obtained by summing all the excitations of interest, thus ensuring in a statistical sense that all needed eigenvectors of the iteration matrix will be generated. In finite precision arithmetic, the algorithm will likely terminate by solving the composite system before many of the solutions corresponding to other right-hand sides have been produced to necessary accuracy. In this situation, the multiple excitation algorithm restarts by using the most recent solutions as the next initial solution estimates, and by iterating directly on the system with the worst error (i.e., using that system to generate the direction vectors employed to expand the remaining systems). The use of the system with the worst error is motivated by the fact that the direction vectors up to this point in the procedure have not spanned that particular solution space well. After the system used as a seed for the direction vectors is solved to desired accuracy, systematic restarting continues until all solutions have been produced to desired accuracy. We often terminated the initial iteration (composite system) at a much lower error level than the other systems in an attempt to initially generate a more complete set of direction vectors.

Because of the systematic restarting, the algorithm remains robust even if specific matrix eigenvectors are absent from the initial seed vector. However, we have found that the use of the composite system as an initial seed significantly reduces the number of required iterations and thus improves the efficiency of the approach. Numerical examples of this multiple excitation algorithm are available in the literature [Smith, Peterson and Mittra 1989].

7.12 Summary

An overview of the numerical implementation of the conjugate gradient algorithm for frequency-domain electromagnetic scattering problems has been discussed. Since the CG algorithm can be used to solve any matrix equation, it can be used in an obvious manner within the numerical treatment of any sort of EM problem. The critical ques-

tion remaining is when to use a direct method of solution and when to use iteration. Situations have been identified where the CG method may be advantageous. In addition, we have discussed the convergence behavior of the algorithm and considered the CG-FFT implementation in detail. This background information is intended to aid the reader in evaluating the suitability of the iterative procedure for a given application.

Acknowledgments

The first author would like to acknowledge helpful discussions with colleagues Paul E. Saylor, Charles F. Smith, and David R. Tanner over the past few years. These individuals have contributed in numerous ways to the author's understanding of the conjugate gradient algorithm.

References

[1] Borup, D. T., and O. P. Gandhi, "Fast-Fourier transform method for calculation of SAR distributions in finely discretized inhomogeneous models of biological bodies," *IEEE Trans. Microwave Theory Tech.*, **MTT-32**, 355–360, April 1984.

[2] Borup, D. T., and O. P. Gandhi, "Calculation of high resolution SAR distribution in biological bodies using the FFT algorithm and conjugate gradient method," *IEEE Trans. Microwave Theory Tech.*, **MTT-33**, 417–419, May 1985.

[3] Brigham, E. O., *The Fast Fourier Transform*, Englewood Cliffs, NJ: Prentice-Hall, 1974.

[4] Catedra, M. F., J. G. Cuevas and L. Nuno, "A scheme to analyze conducting plates of resonant size using the conjugate gradient method and the fast Fourier transform," *IEEE Trans. Antennas Propagat.*, **AP-36**, 1744–1752, Dec. 1988.

[5] Chan, C. H., "A numerically efficient technique for the method of moments solution of electromagnetic problem associated with planar periodic structures," *Microwave and Optical Technology*

Letters, **1**, 372–374, Dec. 1988.

[6] Cwik, T. A., and R. Mittra, "Scattering from frequency selective screens," *Electromagnetics*, **5**, 263–283, 1985.

[7] Daniel, S. M., and R. Mittra, "An optimal solution to a scattering problem," *Proc. IEEE*, **58**, 270–271, 1970.

[8] Davidson, D. B. and D. A. McNamara, "Comparison of the application of various conjugate gradient algorithms to electromagnetic radiation from conducting bodies of revolution," *Microwave and Optical Technology Letters*, **1**, 243–246, Sep. 1988.

[9] Evans, D. J., *Preconditioning Methods: Analysis and Application*, New York: Gordon and Breach, (ed) 1983.

[10] Golub, G. H., and C. F. Van Loan, *Matrix Computations. Baltimore:* The Johns Hopkins University Press, 1983.

[11] Harrington, R. F., *Field Computation by Moment Methods*, Malabar, FL: Krieger, 1982.

[12] Hestenes, M. R., and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *J. Res. Nat. Bur. Stand.*, **49**, 409–435, 1952.

[13] Jennings, A., "Influence of the eigenvalue spectrum on the convergence rate of the conjugate gradient method," *J. Inst. Math. Appl.*, **20**, 61–72, 1977.

[14] Kas, A., and E. L. Yip, "Preconditioned conjugate gradient methods for solving electromagnetics problems," *IEEE Trans. Antennas Propagat.*, **AP-35**, 147–152, Feb., 1987.

[15] Kastner, R., and R. Mittra, "A spectral-iteration technique for analyzing scattering from arbitrary bodies, Part I: Cylindrical scatterers with E-wave incidence," *IEEE Trans. Antennas Propagat.*, **AP-31**, 499–506, May 1983. (a)

[16] Kastner, R., and R. Mittra, "A spectral-iteration technique for analyzing scattering from arbitrary bodies, Part II: Conducting cylinders with H-wave incidence," *IEEE Trans. Antennas Propagat.*, **AP-31**, 535–537, May 1983. (b)

[17] Kastner, R., and R. Mittra, "A new stacked two-dimensional spectral iterative technique for analyzing microwave power deposition in biological media," *IEEE Trans. Microwave Theory Tech.*,

MTT-31, 898–904, Nov., 1983. (c)

[18] Ko, W. L., and R. Mittra, "A new approach based upon the combination of integral equation and asymptotic techniques for solving electromagnetic scattering problems," *IEEE Trans. Antennas Propagat.*, **AP-25**, 187–197, March 1977.

[19] Mittra, R., and C. H. Chan, "Iterative approaches to the solution of electromagnetic boundary value problems," *Electromagnetics*, **5**, 123–146, 1985.

[20] Mittra, R., C. H. Chan and T. Cwik, "Techniques for analyzing frequency selective surfaces – A review," *Proc. IEEE*, **76**, 1593–1615, Dec. 1988.

[21] Montgomery, J. P., and K. R. Davey, "The solution of planar periodic structures using iterative methods," *Electromagnetics*, **5**, 209–235, 1985.

[22] Nyo, H. L., A. T. Adams, and R. F. Harrington, "The discrete convolution method for electromagnetic problems," *Electromagnetics*, **5**, 191–208, 1985.

[23] Oppenheim, A. V., and R. W. Schafer, *Digital Signal Processing*, Englewood Cliffs: Prentice-Hall, 1975.

[24] Pearson, L. W., "A technique for organizing large moment calculations for use with iterative solution methods," *IEEE Trans. Antennas Propagat.*, **AP-33**, 1031–1033, Sept., 1985.

[25] Peters, T. J., and J. L. Volakis, "Application of a conjugate gradient FFT method to scattering from thin material plates," *IEEE Trans. Antennas Propagat.*, **AP-36**, 518–526, April 1988.

[26] Peterson, A. F., "An analysis of the spectral iterative technique for electromagnetic scattering from individual and periodic structures," *Electromagnetics*, **6**, 255–276, 1986.

[27] Peterson, A. F., "Iterative methods: When to use them for computational electromagnetics," *Applied Computational Electromagnetics Society (ACES) Newsletter*, **2**, 43–52, May 1987.

[28] Peterson, A. F., "A comparison of integral, differential and hybrid methods for TE-wave scattering from inhomogeneous dielectric cylinders," *Journal of Electromagnetic Waves and Applications*, **3**, 87–106, 1989.

[29] Peterson, A. F., and R. Mittra, "Method of conjugate gradients for the numerical solution of large body electromagnetic scattering problems," *J. Opt. Soc. Amer. A*, **2**, 971–977, June 1985.

[30] Peterson, A. F., and R. Mittra, "Convergence of the conjugate gradient method when applied to matrix equations representing electromagnetic scattering problems," *IEEE Trans. Antennas Propagat.*, **AP-34**, 1447–1454, Dec., 1986.

[31] Peterson, A. F., and R. Mittra, "Iterative-based computational methods for electromagnetic scattering from individual or periodic structures," *IEEE J. Oceanic Engineering*, Special Issue on Scattering, **OE-12**, 458–465, April 1987.

[32] Peterson, A. F., C. F. Smith, and R. Mittra, "Eigenvalues of the moment method matrix and their effect on the convergence of the conjugate gradient method," *IEEE Trans. Antennas Propagat.*, **AP-36**, 1177–1179, August 1988.

[33] Pries, D. H., "The Toeplitz matrix: Its occurence in antenna problems and a rapid inversion algorithm," *IEEE Trans. Antennas Propagat.*, **AP-20**, 204–206, March 1972.

[34] Rao, S. M., D. R. Wilton and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Trans. Antennas Propagat.*, **AP-30**, 409–419, May 1982.

[35] Ray, S. L., and A. F. Peterson, "Error and convergence in numerical implementations of the conjugate gradient method," *IEEE Trans. Antennas Propagat.*, **AP-36**, 1824–1827, Dec., 1988.

[36] Richmond, J. H., "Scattering by a dielectric cylinder of arbitrary cross section shape," *IEEE Trans. Antennas Propagat.*, **AP-13**, 334–341, May 1965.

[37] Sarkar, T. K., E. Arvas, and S. M. Rao, "Application of the fast Fourier transform and the conjugate gradient method for efficient solution of electromagnetic scattering from both electrically large and small conducting bodies," *Electromagnetics*, **5**, 99–122, 1985.

[38] Sarkar, T. K., and S. M. Rao, "An application of the conjugate gradient method for the solution of the electromagnetic scattering from arbitrarily oriented wire antennas," *IEEE Trans. Antennas Propagat.*, **AP-32**, 398–403, April 1984.

[39] Sarkar, T. K., K. R. Siarkiewicz, and R. F. Stratton, "Survey of

numerical methods for solution of large systems of linear equations for electromagnetic field problems," *IEEE Trans. Antennas Propagat.*, **AP-29**, 847–856, Nov., 1981.

[40] Smith, C. F., A. F. Peterson, and R. Mittra, "A conjugate gradient algorithm for the treatment of multiple incident electromagnetic fields," *IEEE Trans. Antennas Propagat.*, **AP-37**, 1490–1493, Nov. 1989.

[41] Smith, C. F., A. F. Peterson, and R. Mittra, "The biconjugate gradient method for electromagnetic scattering," *IEEE Trans. Antennas Propagat.*, **AP-38**, 938–940, June, 1990.

[42] Stiefel, E. L., "Kernel polynomials in linear algebra and their numerical applications," *Nat. Bur. Stand. Appl. Math. Ser.*, **49**, 1–22, 1958.

[43] Su, C. C., "Calculation of electromagnetic scattering from a dielectric cylinder using the conjugate gradient method and FFT," *IEEE Trans. Antennas Propagat.*, **AP-35**, 1418–1425, Dec. 1987.

[44] Tsao, C. H., and R. Mittra, "A spectral iteration approach for analyzing scattering from frequency selective surfaces," *IEEE Trans. Antennas Propagat.*, **AP-30**, 303–308, March 1982.

[45] van den Berg, P. M., "Iterative computational techniques in scattering based upon the integrated square error criterion," *IEEE Trans. Antennas Propagat.*, **AP-32**, 1063–1071, Oct. 1984.