ON THE USE OF HYBRID FULL-WAVE CG-FFT AND SPECTRALLY PRECONDITIONED CG-FFT METHODS FOR ANALYZING LARGE MICROSTRIP ANTENNA ARRAYS

By

YUAN ZHUANG
B.Eng. , M. Eng.

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Doctor of Philosophy

McMaster University
January, 1996

DOCTOR OF PHILOSOPHY (1996)
(Electrical and Computer Engineering)

MCMASTER UNIVERSITY
Hamilton, Ontario

TITLE: **ON THE USE OF HYBRID FULL-WAVE CG-FFT AND SPECTRALLY PRECONDITIONED CG-FFT METHODS FOR ANALYZING LARGE MICROSTRIP ANTENNA ARRAYS**

AUTHOR: Yuan Zhuang
B.Eng. , (Tsinghua University)
M. Eng., (Dalian University of Science and Technology)

SUPERVISOR(S): Dr. John Litva
Professor, Department of Electrical and Computer Engineering
Director, Communications Research Laboratory
McMaster University

NUMBER OF PAGES: xvii, 220

ON THE USE OF HYBRID FULL-WAVE CG-FFT AND SPECTRALLY PRECONDITIONED CG-FFT METHODS FOR ANALYZING LARGE MICROSTRIP ANTENNA ARRAYS

ABSTRACT

In this thesis, we investigate full-wave hybrid Conjugate Gradient - Fast Fourier Transform (CG-FFT) methods for analyzing microstrip arrays. In particular, we investigate spectral preconditioning with the CG-FFT methods.

First, a general scheme is introduced for implementing CG-FFT using the spatial discretization. In comparison with other schemes, the proposed one yields greater accuracy and has higher efficiency because no finite difference approximations are involved, all the aliasing and truncation errors are eliminated and the size of the zero padded region is kept to a minimum.

A hybrid full-wave CG-FFT method is then developed for analyzing microstrip structures. It combines the proposed CG-FFT scheme and the full-wave complex discrete image technique. With this combination, the spatially discrete scheme is realized which can be used for microstrip structure analysis without losing any full-wave information, while at the same time, measuring only minor computational cost and errors. Therefore the merits of the proposed scheme are extended to microstrip problems straightaway.

To further improve the rate of convergence for the CG-FFT method, a new highly efficient spectrally preconditioned CG-FFT method is introduced. This technique takes full advantage of the FFT by constructing the special forms of preconditioners and performing the preconditioning in the frequency domain. It uses no any additional memory and only O(NlogN) additional operations for constructing and storing the preconditioner. At the same time it has superior convergence properties compared with the conventional CG-FFT method and other existing preconditioned methods.

The hybrid CG-FFT algorithm developed in this thesis is easily interfaced to UNIX simulation software, including a draw editor and an automatic mesh generator. Using this software, we carry out numerical analyses of different types of microstrip antenna arrays. The analyses are corroborated by experimental measurements. A number of array parameters and boundary effects are studied. These are of interest to antenna design engineers, which include: (1) effect of the finite size of arrays, (2) effect of array shape, (3) spurious radiation from the array feed structures, (4) current distributions on the elements, (5) input impedance and (6) array radiation patterns. Conclusion and discussion are addressed. The performance capabilities of the hybrid full-wave CG-FFT method are demonstrated by a modeling study of very large microstrip reflectarrays. Proposals are made for an improved design for the large array, based on the results of the simulations.

ACKNOWLEDGMENT

The author would like to express his gratitude to his advisor, Professor John Litva. He is very grateful to have worked under Dr. Litva's direction and he sincerely appreciates his guidance, encouragement and support.

The author would also like to deeply thank Dr. Ke-Li Wu, member of the committee. The author has benefited very much from his guidance, technical insight, and time towards this thesis. The author also wishes to thank Dr. Anthony Peirce, member of the committee, for his guidance and suggestions to this thesis, especially to the introduction of the preconditioned method.

The author is also indebted to Mr. Chen Wu for his many comments, suggestions and assistance. He would like to thank other professors, research staff and students in the Wireless and Antenna Group and the Communications Research Laboratory (CRL) for their discussions and assistance.

The author is particularly grateful to his parents, and especially to his wife, Lynda, and their lovely daughter, Sherry, for their continued love, support, and patience. He also would like to take this opportunity to thank Lynda for developing the automatic mesh generator for this project. Without it, the simulations would be a nightmare.

Contents

List of Tables

List of Figures

Part I

The Hybrid CG-FFT Method

Chapter 1

Introduction

1.1 Background

Microstrip antennas have been the focus of considerable interest by the antenna community in recent years. A great deal of work has been carried out, both in developing available models and in developing techniques for designing microstrip array antennas.[1] [2]

The concept of a microstrip antenna was first introduced in the 1950's [3]. This type of antenna began to attract more and more attention from researchers after the 1970's. The basic configuration of a microstrip antenna is a metallic layer printed on a thin dielectric substrate with a metallic ground plane. The antenna can be fed in different ways, such as with a microstrip line from the side, coaxial cable or coupling aperture from the back.

The advantages of microstrip antennas include low-profile, low-weight, low-cost, ease of integration into arrays or with microwave integrated circuits (MIC) and polarization diversity. These merits give microstrip antennas promising applications in both the military and the commercial world.

The disadvantages of microstrip antennas include narrow bandwidth, spurious radiation from feed structures, poor polarization purity, limited power capacity, etc. Another disadvantage is that mathematical models of the microstrip antennas have not yet reached a state of maturity, and thus there are few robust and versatile analytical models available for design engineers. In order to satisfy demanding system requirements, much of the recent work has gone into developing ways of overcoming these problems. These studies include the development of novel microstrip antenna configurations and the development of accurate and versatile analytical models for understanding, and furthermore, for designing and optimizing the performance of microstrip antennas.

The objective of the study of microstrip antennas undertaken in this thesis is to develop an efficient and robust model for simulating large microstrip arrays. We will first explore and then develop the hybrid Conjugate Gradient - Fast Fourier Transform (CG-FFT) method. This technique will be shown to be versatile, accurate and efficient. Further, it will be shown that this technique can be used for studying large microstrip planar structures; in particular, large microstrip arrays. The demonstration of the capability of the hybrid CG-FFT will be carried out by applying it to several types of microstrip antenna arrays. Finally , a new spectrally preconditioned CG-FFT method will be presented. It will be shown to provide a considerable improvement in the convergence performance of the conventional CG-FFT method.

Before we proceed with the details, we shall carry out a review of the analytical techniques used in modeling microstrip antennas in Section 1.2 and 1.3.

A description of the scope of the chapters will be given in section 1.4.

1.2 Methods Used for Microstrip Antenna Analysis

With a good antenna model one should be able to calculate all the necessary parameters of the antenna (impedance, gain, pattern, etc.) The simulation results should be accurate enough for design purposes, while at the same time, the procedure for arriving at the solution should be as simple as possible.

Before computers were as widespread as they are today, antenna analysis was limited to geometries that were amenable to analytical solutions. With the advent of easy access to computers, software programs have been developed which provide fairly accurate simulations of wire, reflector, horn and waveguide slot antennas. Compared to these types of antennas, the analysis of microstrip antennas is much more complicated, and still needs improvement.

The microstrip antenna models that have been developed to date can be classified into two categories; which are: (1) simplified, or empirical models and (2) full-wave analytical models.

The Empirical Models maintain simplicity at the expenses of accuracy or versatility. Phenomena such as surface wave propagation and dispersion are generally not included in these models. They can yield reasonable results at microwave frequencies but completely unacceptable results in the millimeter-wave band. Within the empirical models, the transmission line model (TLM) and the cavity model (CM) are the most useful.[4] [5] [6]

The full-wave analytical models numerically carry out an exact solution of Maxwell's equations without the use of approximations as in the case of empirical models. Some features of full-wave analyses that are of interest include:

- accuracy: full wave analyses provide accurate results when used to calculate input impedance , mutual coupling, RCS, etc.
- completeness: full-wave solutions include the effects of the surface wave, space wave radiation and external coupling.
- versatility: full-wave solutions can be applied to microstrip elements and arrays with arbitrary shapes, various types of feeding structures, multilayer geometries and different types of substrates.
- computational complexity: full-wave analyses are numerically intensive and require careful programming so as to be computationally accurate and efficient.

They can generally be categorized into two basic groups. (1) differential equation methods; (2) integral equation methods.

The *Finite-Difference Time-Domain (FDTD) Method* [7] is a differential equation method. In this method, both the temporal and spatial partial derivatives in Maxwell's equations are approximated by finite differences and the resultant equation is solved to find the field distribution as a function of space and time. The advantage of this method is that the algorithm's implementation is very straightforward and there is no need to invert large matrices as is generally the case for integral equation methods. Also, one calculation yields a response over a very large frequency range. However, because the technique solves for both the electric and magnetic fields in a three dimensional space, the number of unknowns is substantially larger than in the case of integral equation methods. In addition, absorbing boundary conditions (ABC), which are used in FDTD to reduce an open-system to a finite-region problem, are only approximate conditions and may lead to unstable solutions. Furthermore, a considerable time sequence of data must be continuously stored.

The algorithm developed in this thesis is based on the integral equation method. In this model, it is usually assumed that the substrate is infinite in extent in the lateral directions. The proper boundary conditions are enforced at the air-dielectric interface. By using the exact Green's function for the dielectric substrate, space wave radiation, surface wave radiation, dielectric loss, and mutual coupling, internally and externally, are all included in the model. Therefore, this model is very accurate and versatile. But the computational cost is high. In this model, the electric or magnetic current distribution on the antenna elements and feeds are the unknowns to be solved instead of the electric or magnetic field components outside of the microstrip as is the case with FDTD modeling. Generally, the microstrip structure consists of very thin metallic surfaces, therefore, the current distributions are located on a 2-D surface, which result in much smaller number of unknowns compared with FDTD. This method generally employs the electric field integral equation (EFIE) or the magnetic field integral equation (MFIE) to solve the unknown current distributions [8] [9]. This is done by expanding the currents by a set of basis functions, then using a set of weighting (test) functions to discretize the integral equation. The key steps in this model are (1) the choice of the basis functions and test functions; (2) the evaluation of the impedance matrix elements.

The evaluation of the impedance matrix elements involves the integration which is given in 2.5. Since the evaluation of this integral involves surface wave poles, a slowly converging integrand, and the possibility of singularities associated with source points, it requires very careful treatment. It is also the most time-consuming part of the solution. Because of the different treatment of the impedance matrix $Z_{m,n}$, the integral equation methods can further be categorized into spatial-domain methods [9] and spectral-domain methods [10].

1.3 Use of Efficient Iterative Methods for EM Analysis

In the conventional integral equation methods, characterized by the Method of Moments (MoM), the discretization procedure that is used to evaluate the integral yields a square matrix (impedance matrix Z). The memory requirements for storing the matrix are of order $O(N^2)$, where N is the number of unknowns. The linear equation is usually solved by standard matrix inversion methods such as Gaussian elimination or LU decomposition. The vital shortcoming of such direct inversion approaches is that the sizes of matrices that can be inverted are limited by the available computer resources (memory size and CPU time). For instance, if 10 segments per wavelength are used to expand the current in each dimension, it needs O(100 Megabytes) of computer internal memory to analyze a $10\lambda \times 10\lambda$ structure. This prohibitively large internal memory requirement makes the traditional matrix inversion approaches unattractive. Therefore it follows that alternative methods must be developed in order to meet the ever increasing demands for simulating large EM structures.

Iterative methods have been used for many years by researchers to address these requirements. The major advantages of iterative methods include:

- Only $O(N)$ computer memory is required for implementing iterative methods. This is because the technique requires only the multiplication of matrices with vectors, there is no need to generate and store the system matrix. This feature makes iterative approaches highly suitable for EM analysis of electrically large scale structures;

- Ill-conditioned linear systems can be solved using iterative methods simply by carrying out more iterations, while the direct inversion methods may totally fail to yield an accurate solution.

In this study, we will explore the application of an iterative algorithm - the Conjugate Gradient - Fast Fourier Transform (CG-FFT) method [18] [20] to the analysis of microstrip structures, in particular, large microstrip antenna arrays. Among the known iterative algorithms, the CG-FFT method is the most popular one for EM analysis. This is primarily because this method ensures the convergence, even for arbitrary initial estimates. Another advantage follows from the convolution property of the integral-differential operators, which involves the induced current density and the pertinent Green's function. By employing the convolution theorem, the operation can be carried out efficiently via the fast Fourier transform (FFT). This practice makes a substantial improvement in the computation speed ($O(N_i N_s log N_s)$ compared to $O(N_s^3)$) for the direct inversion approach. Here, N_s is the number of unknowns, N_i is the number of iterations for the CG algorithm which is less than N_s.

In the conventional CG-FFT method [17], the analytical form of the Fourier transform of the Green's function is used when implementing the convolution theorem. This procedure requires that extensive zero padding be used with the FFT in order to reduce aliasing errors caused by the fact that the spectral Green's function is usually widely spread in the frequency domain. To overcome this difficulty, a spatially discrete scheme has been proposed[11]. In this scheme, the integral equation is converted to a discrete periodic form and the spatial derivatives are approximated using finite differences. The discrete circular convolution then is calculated using the Discrete Fourier Transform (DFT). The advantage of this scheme is that the extent of the zero padding is reduced and the aliasing errors are eliminated.

When applying the CG-FFT method to the analysis of microstrip structures, most schemes proposed so far use the conventional CG-FFT approach [12] [13] instead of the spatially discrete approach. The reason is because the spatially discrete CG-FFT scheme needs to use the spatial Green's function that is applicable to the microstrip structures which are to be analyzed. The involved Sommerfeld integral is a very cumbersome obstacle in the way of reaching this objective. The conventional CG-FFT method uses the analytical form of the spectral domain Green's function which is readily available for circumventing the difficulties associated with solving the Sommerfeld integral. However, alias errors are inevitable when the conventional approach is adopted. The only way to overcome these errors is to first obtain the closed form spatial Green's function, and then apply the spatially discrete CG-FFT scheme [22] [23].

In this thesis, a hybrid analysis technique [22] is presented, which consists of the spatially discrete CG-FFT method combined with the complex discrete image technique [24] (CDIT in short). This approach can realize the spatially discrete CG-FFT scheme for the analysis of microstrip problems. The CDIT is used to derive the closed form spatial Green's function for a microstrip structure, with negligible computational overhead. The integral equation describing the microstrip problem is discretized in the spatial domain . The spatially discrete CG-FFT is then applied to the resultant system. This scheme overcomes difficulties associated with the numerical evaluation of the Sommerfeld integral when deriving the spatial Green's function. At the same time, it achieves the high efficiency and accuracy by avoiding the aliasing error existing in other CG-FFT schemes that are used for analyzing microstrip structures.

Another major contribution of this thesis is the development of the spectrally preconditioned CG-FFT method for microstrip analysis. As we know, the convergence rate of the CG method depends on the spectrum of the system matrix (impedance matrix in our case). Generally speaking, the CG method converges faster if the matrix has a small condition number or clustered eigenvalues [14] [15] [16]. In order to accelerate the CG convergence rate, a preconditioning step can be introduced at each CG iteration, which leads to the preconditioned conjugate gradient (PCG) method. In this study, an efficient preconditioned method - the spectrally preconditioned CG-FFT (SPCG-FFT) method is developed for applications in electromagnetic analysis. The technique fully exploits the advantages to be gained from using the FFT in this analysis by bringing the CG operation, as well as the development and implementation of the preconditioner, into the spectral domain. It uses all of the information that is generated during the course of the CG procedure. The algorithm requires no additional memory and only $O(N \log N)$ operations for its implementation. Compared with the non-preconditioned CG-FFT method, the SPCG-FFT is found to have superior convergence performance, which makes this technique a very promising alternative iterative method for the analysis of large microwave structures.

1.4 Organization of the chapters

The thesis is divided into two parts. Part I, which includes Chapter 2 to Chapter 4, deals with the hybrid CG-FFT method and its application to the analysis of large microstrip array antennas. In Part II, i.e. Chapter 5 and Chapter 6, are presented the development of a new spectrally preconditioned CG-FFT method for electromagnetic analysis.

In Chapter 2 of Part I, we start with a brief introduction of the CG-FFT method. This is followed by a detailed description of the spatial domain discrete CG-FFT scheme. The chapter concludes with a comparison of the performance of the conventional CG-FFT and spatially discrete CG-FFT methods.

Details dealing with the implementation of the spatially discrete CG-FFT for solving microstrip problems are discussed in Chapter 3. The use of the complex discrete image technique for analyzing microstrip structures is introduced in this chapter. This is followed by a description of a hybrid technique, composed of the CDIT and the spatially discrete CG-FFT technique. Some numerical comparisons are made between the proposed and other CG-FFT schemes in the last portion of this chapter.

In Chapter 4, the hybrid CG-FFT technique is applied to analyzing a series of large microstrip arrays, including, microstrip dipole arrays, microstrip planar arrays and microstrip reflectarrays [54] [67] [68] [69]. A study is made of the edge effect, shape effect and feed network effect on the array current distributions, radiation patterns and input impedances.

The spectrally preconditioned CG-FFT algorithm is presented in Chapter 5 of Part II. The concept of preconditioning is introduced first. The preconditioning procedure is interpreted by means of linear system theory. The techniques for implementing the preconditioners and carrying out preconditioning operations in the spectral domain are presented. A systematic approach for designing a large class of spectral preconditioners is presented.

In Chapter 6, the spectral properties of the proposed spectral preconditioners are studied and compared with other types of preconditioners. The advantages of using preconditioning to improve convergence rates are discussed using numerical examples.

Conclusions which follows from this study are presented in Chapter 7.

Chapter 2

The Spatially Discrete CG-FFT Method

2.1 Introduction

The CG-FFT method is used in this thesis to solve the unknown current distributions on the elements of a microstrip antenna. This is carried out via the electric field integral equation formulation of the problem. In the integral equation model, the unknown current is related to the electric field by means of the Green's function in the following typical form,

$$\int_s \overline{\overline{G}}(x, y; x', y') \cdot \vec{J}(x', y') dx' dy' = -\vec{E}_{inc}(x, y) \tag{2.1}$$

where $\overline{\overline{G}}(x, y; x', y')$ is the dyadic Green's function, $\vec{E}_{inc}(x, y)$ is the incident field and $\vec{J}(x', y')$ is the unknown current distribution on S as shown in Fig.2.2.

To solve the above integral equation, one first expands the unknown current using a set of expansion (basis) functions

$$\vec{J}(x,y) = \sum_{n=1}^{N} I_n \vec{f}_n(x,y), \tag{2.2}$$

where $\vec{f}_n(x,y)$ and I_n are the basis functions and the expansion coefficients, respectively. Substituting the expanded form of the current into 2.1, we have

$$\sum_{n=1}^{N} I_n \int_{x'} \int_{y'} \overline{\overline{G}}(x,y;x',y') \cdot \vec{f}_n(x',y')dx'dy' = -\vec{E}_i(x,y). \tag{2.3}$$

The next step is to discretize the integral equation by carrying out the inner products on both sides of 2.3 with a set of weighting (test) functions $\vec{t}_m(x,y), m = 1,2,3,\ldots$ This procedure yields a linear system as follows:

$$\sum_{n=1}^{N} Z_{m,n} I_n = V_m; \qquad m = 1,2,\ldots,N \tag{2.4}$$

$$Z_{m,n} = \int_x \int_y \left[\int_{x'} \int_{y} \overline{\overline{G}}(x,y;x',y') \cdot \vec{f}_n(x',y')dx'dy'\right] \vec{t}_m(x,y)dxdy \tag{2.5}$$

$$V_m = -\int_x \int_y \vec{E}_{inc}(x,y)\vec{t}_m(x,y)dxdy. \tag{2.6}$$

The key steps in this method are: (1) the choice of basis functions and test functions; and (2) the evaluation of the elements of the impedance matrix $Z_{m,n}$.

In the conventional moment method, the linear set of equations is solved using a matrix inversion approach, such as Gaussian elimination. However, as the size of the problem increases (the number of unknowns increases), the computer memory and computing time increase rapidly ($O(N^2)$ and $O(N^3)$, respectively), which restricts the use of the method to simple and small-scale problems.

Iterative methods are more powerful for the solution of large systems. This is because iterative methods only involve the multiplication of matrices with vectors and do not require an explicit storage of the system matrix, and they also avoid the requirement for inverting a matrix, which is subject to numerical instability for ill-conditioned matrices.

2.2 The Conjugate Gradient Method

The conjugate gradient (CG) method is an iterative conjugate-direction optimization algorithm. It converges to the minimum of a function f for a vector X. This algorithm depends on two subsidiary calculations: (i) calculation of the gradient of the function $\Delta f(X)$ at an arbitrary point, and (ii) minimization of f along a specific ray, i.e., determination of the value λ that minimizes the expression $f(X+\lambda u)$ for specified X and u.

To start the development, we define a function

$$f(X) \equiv \frac{1}{2}|AX - b|^2 \tag{2.7}$$

where b is an N element vector and A is an $N \times N$ matrix. This function has only one single minimum, which is achieved by choosing X that satisfies the linear equation

$$AX = b. \tag{2.8}$$

Thus, minimizing the function $f(X)$ and solving the linear equation $AX = b$ are equivalent problems. The conjugate gradient method can therefore be used to solve a linear equation.

To solve the functional problem, one minimizes the residual of the linear equation, i.e.,

$$R = AX - b. \tag{2.9}$$

Using the gradient information, the convergence of the minimization process is accomplished via a systematic orthogonalization of the solution vector with respect to the residual vector at the end of each iteration. The procedure starts out with an initial estimate X_0 and a corresponding residual error $R_0 = AX_0 - b$. The CG algorithm provides a prescription for updating the set of directions for minimizing the residual R in such a way that the successive direction vectors $\{P_n\}$ are mutually A-orthogonal or conjugate, i.e.,

$$< P_i, AP_i >= 0 \quad i \neq j. \tag{2.10}$$

The set of mutually conjugate directions guarantees that the minimization along one direction is not "spoiled" by the subsequent minimization along another, so that interminable cycling through a set of directions can be avoided. Consequently, for a system representing N unknowns, the exact solution is obtained at the N-th iteration with convergence being monotonic throughout the whole process.

There are different versions of the CG algorithm. The one we followed is as follows [18]:

$$
\begin{array}{l}
k = 0; \quad x_0 = 0; \quad r_0 = b \\
while \quad r_k \neq 0 \\
\quad k = k + 1 \\
\quad if \;\; k = 1 \\
\qquad\qquad p_1 = A^a r_0 \\
\quad\;\; else \\
\qquad\qquad \beta_k = \dfrac{||A^a r_{k-1}||^2}{||A r_{k-2}||^2} \\
\qquad\qquad p_k = A^a r_k + \beta_k p_k \\
\quad\;\; end \\
\quad\;\; \alpha_k \quad = \dfrac{||A^a r_{k-1}||^2}{||A p_k||^2} \\
\quad\;\; x_k \quad = x_{k-1} + \alpha_k p_k \\
\quad\;\; r_k \quad = r_{k-1} - \alpha_k A p_k \\
end
\end{array}
$$

$$x = x_k,$$

$$(2.11)$$

where the norm and the adjoint operators are defined in terms of inner products

$$||U||^2 =< U, U >, \tag{2.12}$$

$$< AU, V >=< U, A^a V > . \tag{2.13}$$

Using this definition, it can be shown that the adjoint operator of 2.1 is

$$A^a[J] = \int\int_{S'} \overline{\overline{G}}^*(|\vec{r} - \vec{r}'|) \cdot \vec{J}(\vec{r}')ds', \tag{2.14}$$

where $*$ denotes the complex conjugate.

When the CG procedure is applied to a problem, it is found that the most time-consuming computations are the operations Ap_n, Ar_n and $A^a r_n$ in the form given by 2.1 and 2.14. It will be shown that these integral operations are of the convolution type and can therefore be calculated efficiently via an FFT by invoking the convolution theorem. The definitions and properties of the Fourier transform are listed in Appendix A.

2.3 The CG-FFT Algorithm and the Continuous Fourier Transform

We consider the problem of scattering from a thin wire antenna (length l and radius $a \ll l$) in free space to illustrate the application of the CG-FFT procedure.

The integral equation relating the incident field E_z^i to the induced current J_z can be derived from 2.1, which is

$$E_z^i(z) = -\frac{1}{j4\pi\omega\epsilon_0}\left(k_0^2 + \frac{d^2}{dz^2}\right)\int_0^l J(z')G(z,z')dz', \tag{2.15}$$

where G(z,z') is the Green's function in free space which is given by

$$G(z,z') = G(z-z') = \frac{e^{-jk_0 r}}{r}, \qquad r = \sqrt{(z-z')^2 + a^2}.$$

It is clear that the integration in 2.15 is the convolution of the current $J(z')$ and the Green's function $G(z,z')$. Invoking the convolution theorem and using a property of the Fourier transform(FT):

$$\mathcal{F}\left[\frac{\partial^n}{\partial x^n} g(x)\right] = (jk_x)^n \tilde{g}(k_x),$$

we have

$$A[J] = -\frac{1}{j4\pi\omega\epsilon_0}\mathcal{F}^{-1}\left\{(k_0^2 - k_z^2)\tilde{G}(k_z)\tilde{J}(k_z)\right\}, \qquad (2.16)$$

where the continuous Fourier transform of the Green's function is

$$\tilde{G}(k_z) = \frac{1}{2\pi}K_0(a\sqrt{k_z^2 - k_0^2}), \qquad (2.17)$$

where K_0 denotes the zeroth order modified Bessel function of the second kind.

Up to this point, the development appears to be totally mathematical. We have the continuous Fourier Transform (FT) of the Green's function and the derivative d^2/dz^2 in the spatial domain has been transferred to a simple algebraic operation in the spectral domain. Whenever the quantity AJ is to be calculated, we simply calculate the FT of J and then do the inverse FT of the product of $\tilde{J}$ and $\tilde{G}$.

However, there are some underlying issues that need to be treated very carefully when implementing these procedures numerically. In the computer calculations, we use the sampled or discrete form of a continuous function. Therefore, the continuous FT needs to be replaced by the DFT and then we use the computationally efficient FFT to calculate the DFT in order to speed up the processing.

It has been shown [19] that when making the transition from the continuous FT to the FFT, some considerable numerical errors can be generated if the continuous Fourier transform of the Green's function is employed in a CG-FFT algorithm.

To implement the operation $A[J_n]$ (J_n is the estimated value of J for the CG algorithm's n-th iteration) via the FFT, the DFT of G and J_n must be obtained first. Since J_n is already a discrete sequence (the expansion coefficients of the current $J(z)$), there is no difficulty in carrying out the DFT of J_n (correspondingly for p_n and r_n in 2.11). But there are difficulties in doing the same for the Green's function. As shown in Fig.2.1, the Green's function has a singularity at $r = 0$, therefore its frequency content is widely spread in spectral space. If $G(z)$ is sampled with the same sampling rate as that used for $J(z)$, a very large error due to aliasing can be generated, which renders the solution of the CG-FFT to be completely unacceptable. In order to reduce this error, the FT of G has to be sampled over a very large range of frequencies, which means that the Green's function, and correspondingly the current distribution, have to be sampled using a very large sampling rate. Thus, the number of sampled points has to be large. This, of course, results in the requirement for a large computer memory and long computational time. There is one technique which can be used to save a little computing time: we start by carrying out the DFT for $J(z)$ using a low sampling rate, then we pad the results with a large number of zeros to make the output match the resultant DFT for $G(z, z')$ obtained with a large sampling rate. There is, however, no reduction in the memory requirement using this approach.

Another shortcoming of this version of the CG-FFT is the treatment of the spatial derivatives of the grad-div operator in the EFIE with singular kernels, especially for 2D and 3D cases. Let us take scattering from a 2D metallic plate in free space as an example. In this case, the differential-integral operation is carried

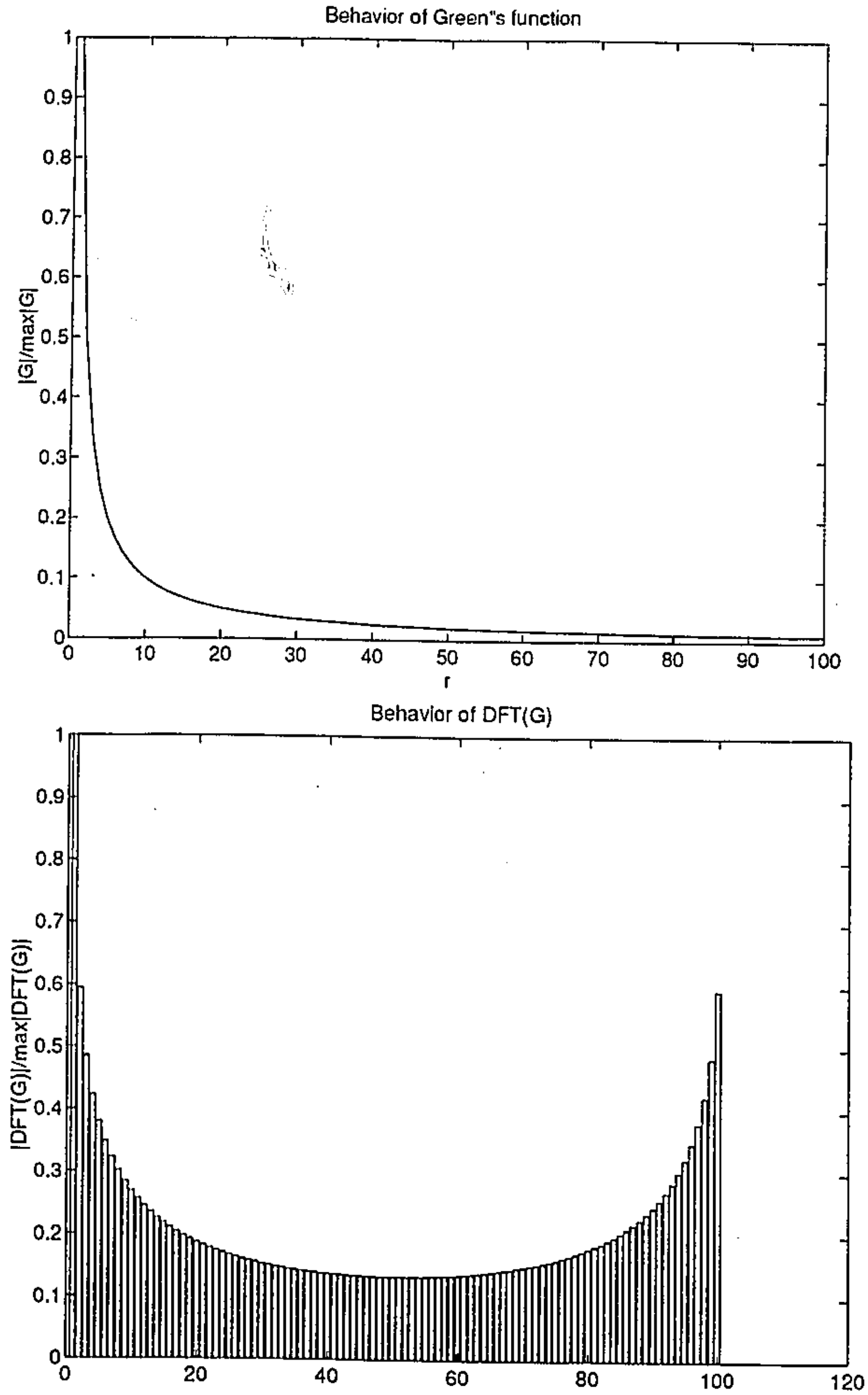


Figure 2.1: The behavior of Green's function in the spatial and spectral domain

out via the FFT as [18]

$$\begin{aligned} A[\tilde{J}] &= \mathcal{F}^{-1}\Big\{k^2[\hat{x}\tilde{J}_x\tilde{G}_0 + \hat{y}\tilde{J}_y\tilde{G}_0] \\ &+ [\hat{x}(jk_x) + \hat{y}(jk_y)][jk_x\tilde{J}_x\tilde{G}_0 + jk_y\tilde{J}_y\tilde{G}_0]\Big\}. \end{aligned} \tag{2.18}$$

The continuous Fourier transform of the Green's function is

$$\tilde{G}_0(k_x, k_y) = \frac{2\pi j}{\sqrt{k_x^2 + k_y^2 - k^2}}. \tag{2.19}$$

The spatial derivatives are replaced by simple algebraic manipulations in the spectral domain. It has been found [25] [26] that this practice leads to substantial numerical difficulties. Even though the global error in the CG iteration is found to decrease monotonically (but very slowly), the simulated distributions of the current on the plate are found to be wrong. Only when a very fine mesh is used, can reasonable results be obtained. The causes for this numerical difficulty can be explained from two different aspects. First, by operating in the spectral domain, the differentiation is carried out over the total area being analyzed. The current density, on the other hand, is not continuously differentiable at the edges of the plate. Second, the global differentiation is applied directly to the integral operator, while the kernel of the integral is singular, even after being transformed into the spectral domain. Both of these factors result in substantial errors, which mean that the simulations do not give an accurate reflection of reality. Therefore, even though the technique converges to give very small global errors, the simulated current distributions do not agree well with measurements.

The large aliasing errors that result from the use of the continuous form of $\tilde{G}$ and the errors introduced by the improper treatment of the $\nabla\nabla\cdot$ operators in the differential-integral equation render this form of the CG-FFT algorithm to be numerically unstable and computationally inefficient. The use of coarse grids can result in very coarse current distributions with a lot of obvious spurious ripples. Furthermore, if one adopts a finer grid in order to get a more reasonable solution, it is found that the requirements for memory and computer time increase rapidly. In the next section, a spatially discrete CG-FFT scheme will be introduced aimed at improving this conventional CG-FFT method.

2.4 The Spatially Discrete CG-FFT Scheme

In this section is presented the spatially discrete CG-FFT algorithm. The derivation is based on a consideration of radiation and scattering from two dimensional structures. The development for other types of applications can be easily obtained by following the procedure described below. The differences between this scheme and others, as well as advantages and disadvantages, will be discussed in the next section.

In the case of the 2D metallic structure shown in Fig.2.2, let S denote the metallic surface and $\hat{n}$ denote the unit vector which is normal to the surface. The electric field integral equation is used in the analysis by first applying the PEC boundary condition to the metal surface (perfect conductor), i.e.:

$$\hat{n} \times (\vec{E}^i(\vec{r}) + \vec{E}^s(\vec{r})) = 0 \qquad \vec{r} \in S \tag{2.20}$$

where $\vec{E}^i$ and $\vec{E}^s$ are the incident and scattered fields. It follows that

$$-\hat{n} \times \vec{E}^i(\vec{r}) = \hat{n} \times (-jw\vec{A}(\vec{r}) - \nabla\Phi(\vec{r})) \qquad \vec{r} \in S \tag{2.21}$$

with the magnetic vector potential defined by

$$\vec{A}(\vec{r}) = \frac{\mu}{4\pi} \int\int_S \vec{J}_s(\vec{r}') \cdot \overline{\overline{G}}_A(\vec{r}, \vec{r}') dS', \tag{2.22}$$

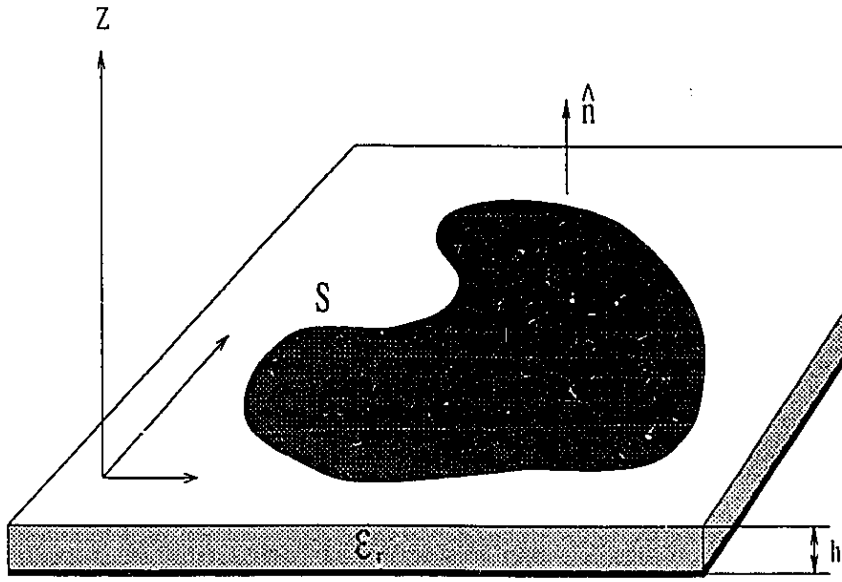


Figure 2.2: The microstrip patch with arbitrary shape.

where $\overline{\overline{G}}(\vec{r},\vec{r}')$ is the dyadic Green's function for the magnetic vector potential and $\vec{J}_s$ is the surface current density and prime (′) denotes the source. The scalar potential due to the charge distribution can be defined by using the Lorentz's gauge:

$$\Phi(\vec{r}) = -\frac{1}{j\omega\mu\epsilon}\int\int_S \left(\nabla\cdot\overline{\overline{G}}_A(\vec{r},\vec{r}')\right)\cdot\vec{J}_s(\vec{r}')dS'. \tag{2.23}$$

The Green's function G_V associated with the scalar potential Φ can be defined by applying the condition of uniqueness of G_V. Thus we can write

$$\nabla\cdot\overline{\overline{G}}_A(\vec{r},\vec{r}') = \mu\epsilon\nabla G_V(\vec{r},\vec{r}') = -\mu\epsilon\nabla' G_V(\vec{r},\vec{r}'). \tag{2.24}$$

Now 2.23 can be expressed as

$$\Phi(\vec{r}) = \frac{1}{j\omega}\int\int_S \nabla' G_V(\vec{r},\vec{r}')\cdot\vec{J}_s(\vec{r},\vec{r}')dS'. \tag{2.25}$$

The next step is to remove the ∇ from the Green's function to the current distribution function. The advantage of this treatment will be discussed in the next section.

Using the vector identity

$$\nabla a \cdot \vec{B} = \nabla \cdot (a\vec{B}) - a\nabla \cdot \vec{B} \tag{2.26}$$

and Gauss' theorem, we have the new form for 2.25 as follows,

$$\Phi(\vec{r}) = -\frac{1}{j\omega}\int\int_S G_V(\vec{r},\vec{r}')\nabla' \cdot \vec{J}_s(\vec{r}')dS' + \oint_{\delta s} G_V(\vec{r},\vec{r}')\vec{J}_s(\vec{r}') \cdot \hat{s}dl \tag{2.27}$$

where δs is the perimeter of the metal surface and $\hat{s}$ is the outward normal unit vector as shown in Fig.2.3. Because there is no current flowing out of the conducting

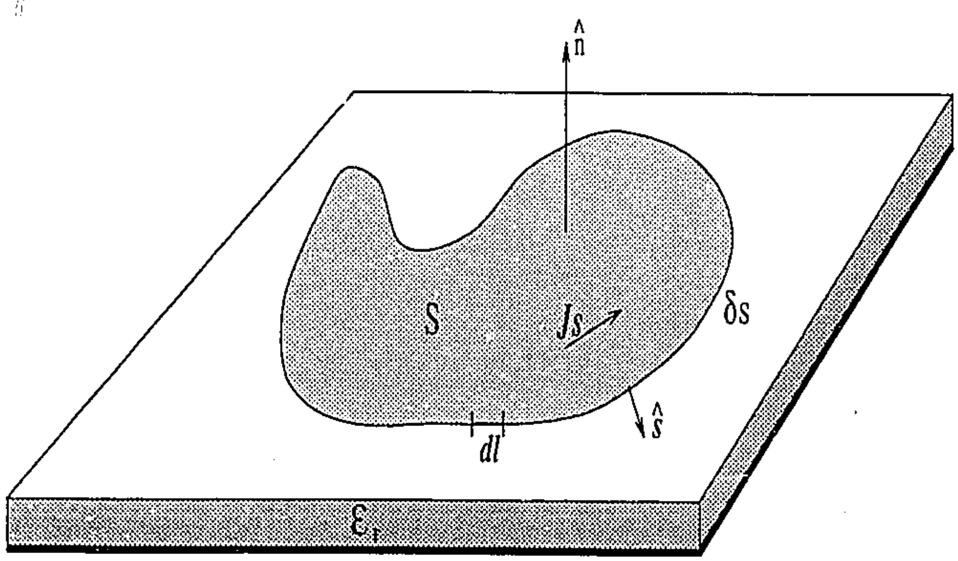


Figure 2.3: Metallic surface corresponding to the last term in (2.27).

surface, $\vec{J}_s \cdot \hat{n} = 0$. Therefore,

$$\Phi(\vec{r}) = -\frac{1}{j\omega}\int\int_S G_V(\vec{r},\vec{r}')\nabla' \cdot \vec{J}_s(\vec{r}')dS'. \tag{2.28}$$

The edge condition guarantees that the normal component of the surface current vanishes on the perimeter of the metal surface. Hence, the last term in 2.27 can be eliminated.

We adopt Galerkin's procedure for solving the integral equation. With the objective of using the FFT for carrying out the surface integration, the patch and its surroundings are circumscribed by a rectangle. The rectangle is then divided into $(M+1) \times (N+1)$ small rectangular cells with sides Δx and Δy along x and y directions, respectively (see Fig.2.4). The arbitrary shape of patch is

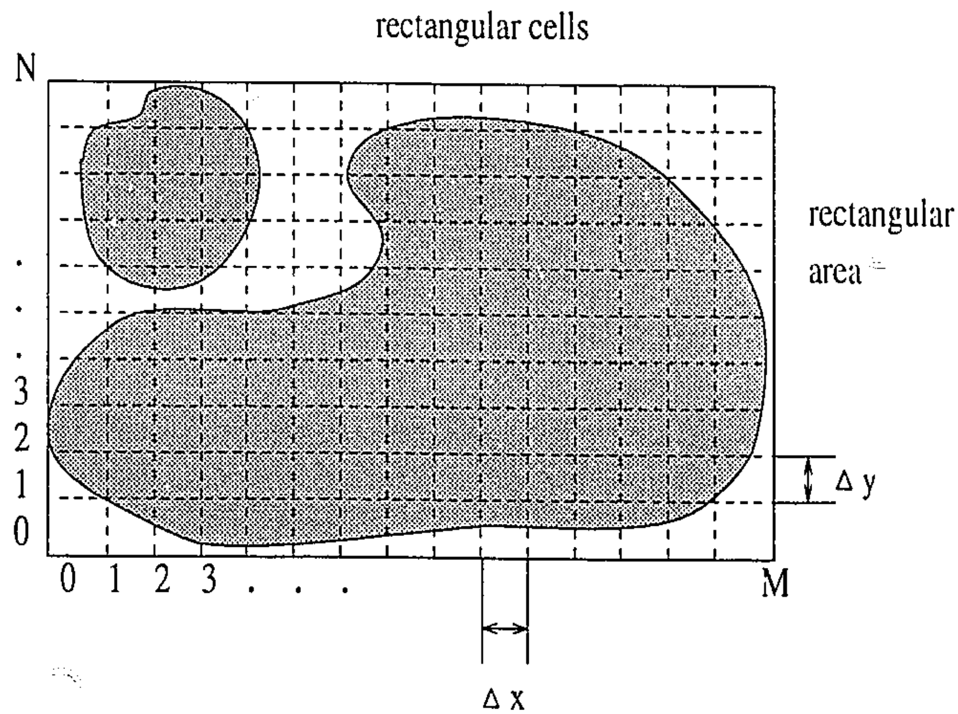


Figure 2.4: The mesh used to discretize the arbitrarily shaped microstrip patch. Note that the border of the meshed region must be rectangular so that we can use the FFT to evaluate the integral.

then approximated by the borders of the small cells which overlap the area of the patch. The well-known roof-top basis function $\vec{f} = f_x^{m,n}\hat{x} + f_y^{m,n}\hat{y}$ [9] is used to expand the current $\vec{J}(\vec{r})$ on S. In Fig.2.5 is shown the shape of the basis functions. Mathematically these functions are

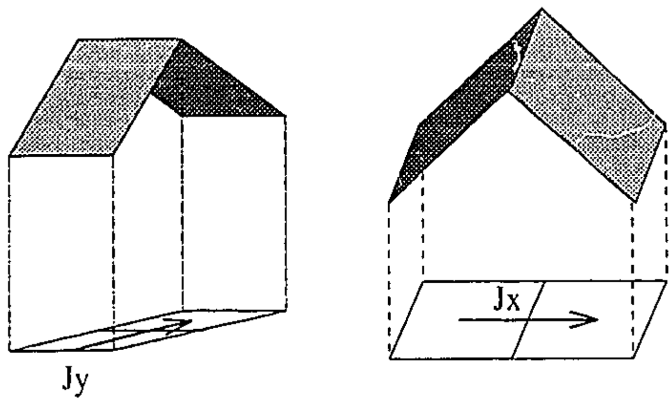


Figure 2.5: The roof-top basis functions used in the expansion of the current distribution in x- and y-directions.

$$f^x_{m,n}(x,y) = \begin{cases} 1 - \frac{|x-m\Delta x|}{\Delta x} & |x - m\Delta x| \leq \Delta x \\ & |y-(n-0.5)\Delta y| < \frac{\Delta y}{2} \\ 0 & elsewhere \end{cases} \tag{2.29}$$

$$f^y_{m,n}(x,y) = \begin{cases} 1 - \frac{|y-n\Delta y|}{\Delta y} & |x-(m-0.5)\Delta x| \leq \frac{\Delta x}{2} \\ & |y - n\Delta y| < \Delta y \\ 0 & elsewhere \end{cases} . \tag{2.30}$$

The derivatives of the basis functions, which will be used later, are

$$\frac{\partial f^x_{m,n}}{\partial x} = \Pi_{m,n}(x,y) - \Pi_{m+1,n}(x,y) \tag{2.31}$$

$$\frac{\partial f^y_{m,n}}{\partial y} = \Pi_{m,n}(x,y) - \Pi_{m,n+1}(x,y), \tag{2.32}$$

where the 2-D unit pulse function defined over the $(i,j)^{th}$ charge expansion segment is

$$\Pi_{m,n}(x,y) = \begin{cases} 1 & |x-(m-0.5)\Delta x| < 0.5\Delta x \\ & |y-(n-0.5)\Delta y| < 0.5\Delta y \\ 0 & elsewhere \end{cases} . \tag{2.33}$$

The surface current is expressed as,

$$\vec{J} = J_{sx}\hat{x} + J_{sy}\hat{y} \tag{2.34}$$

where

$$\begin{aligned} J_{sx}\hat{x} &= \sum_{m=1}^{M}\sum_{n=1}^{N+1} I_x(m,n) f_x^{m,n}(x,y), \\ J_{sy}\hat{y} &= \sum_{m=1}^{M+1}\sum_{n=1}^{N} I_y(m,n) f_y^{m,n}(x,y). \end{aligned} \tag{2.35}$$

From the above expression, it is readily found that

$$\frac{\partial J_{sx}}{\partial x} = \sum_{m=1}^{M+1}\sum_{n=1}^{N+1} \frac{I_x(m,n) - I_x(m-1,n)}{\Delta x} \Pi_{m,n}(x,y) \tag{2.36}$$

$$\frac{\partial J_{sy}}{\partial y} = \sum_{m=1}^{M+1}\sum_{n=1}^{N+1} \frac{I_y(m,n) - I_y(m,n-1)}{\Delta y} \Pi_{m,n}(x,y). \tag{2.37}$$

Here it is implied that

$$\left.\begin{aligned} I_x(0,n) &= 0 \\ I_x(M+1,n) &= 0 \end{aligned}\right\} \quad for \;\; n = 1, ..., N+1$$

$$\left.\begin{aligned} I_y(m,0) &= 0 \\ I_x(m,N+1) &= 0 \end{aligned}\right\} \quad for \;\; m = 1, ..., M+1.$$

If the inner product is defined as

$$< \xi, \eta > = \int_S \xi \cdot \eta^* dS, \tag{2.38}$$

it follows from the Galerkin's procedure that

$$< \vec{E}^s, \vec{f} > = j\omega < \vec{A}, \vec{f} > + < \nabla\Phi, \vec{f} > . \tag{2.39}$$

Using the same vector identity as in 2.27, as well as Gauss' theorem, the last term can be written as,

$$< \nabla\Phi, \vec{f} > = \int\int_S \nabla\Phi \cdot \vec{f} dS = -\int\int_S (\nabla \cdot \vec{f})\Phi dS. \tag{2.40}$$

Up to this point, all of the ∇ operators have been removed from the singular Green's function and are applied to the common basis and testing functions.

Without any loss of generality, 2.39 can be written in Cartesian coordinates as

$$
\begin{aligned}
&\hat{x}\int\int_S E_x^s f_x dS + \hat{y}\int\int_S E_x^s f_y dS = \\
&\quad \hat{x}\left\{\frac{j\omega\mu}{4\pi}\int\int_S f_x\left[\int\int_S J_{sx} G_{Ax}(x-x',y-y')dS'\right]dS\right. \\
&\quad \left. +\frac{1}{j4\pi\epsilon\omega}\int\int_S \frac{\partial f_x}{\partial x}\left[\int\int_S\left(\frac{\partial J_{sx}}{\partial x'}+\frac{\partial J_{sy}}{\partial y'}\right)G_V(x-x',y-y')dS'\right]dS\right\}+ \\
&\quad \hat{y}\left\{\frac{j\omega\mu}{4\pi}\int\int_S f_y\left[\int\int_S J_{sy} G_{Ay}(x-x',y-y')dS'\right]dS\right. \\
&\quad \left. +\frac{1}{j4\pi\epsilon\omega}\int\int_S \frac{\partial f_y}{\partial y}\left[\int\int_S\left(\frac{\partial J_{sx}}{\partial x'}+\frac{\partial J_{sy}}{\partial y'}\right)G_V(x-x',y-y')dS'\right]dS\right\}. \quad (2.41)
\end{aligned}
$$

After replacing the J_{sx} and J_{sy} with their expanded forms, we have

$$
\begin{aligned}
L\vec{J} &= LJ_{sx}\hat{x} + LJ_{sy}\hat{y} \\
&= E_{xd}(i,j)\hat{x} + E_{yd}(i,j)\hat{y} && (2.42)\\
LJ_{sx} &= \frac{1}{\Delta x}\sum_{i'=1}^{N+1}\sum_{j'=1}^{M+1}\left\{\frac{c_1}{\Delta x}[I_x(i',j') - I_x(i'-1,j')]\right. \\
&+ \left.\frac{c_1}{\Delta y}[I_y(i',j') - I_y(i',j'-1)]\right\} \\
&\quad [\Gamma_\Pi(i+1-i',j-j') - \Gamma_\Pi(i-i',j-j')] \\
&- c_2\sum_{i=1}^{N}\sum_{j=1}^{M+1} I_x(i',j')\Gamma_{Ax}(i-i',j-j') && (2.43)\\
\Gamma_{Ax}(i,j) &= \int\int_S f_x^{i,j}\left[\int\int_S f_x^{0,0} G_{Ax}(x-x',y-y')dS'\right]dS && (2.44)\\
\Gamma_\Pi(i,j) &= \int\int_S \Pi^{i,j}\left[\int\int_S \Pi^{0,0} G_V(x-x',y-y')dS'\right]dS && (2.45)\\
E_{xd}(i,j) &= \int\int_S f_x^{i,j} E_x^{inc}(x,y)dS, && (2.46)
\end{aligned}
$$

where $c_1 = 1/4\pi\epsilon\omega and c_2 = j\omega\mu/4\pi$. The quantities $LJ_{sy}(i,j)$, $\Gamma_{Ay}(i,j)$ and $E_{yd}(i,j)$ can be obtained in a similar way.

We have now reached the point where the continuous integral equation has been completely discretized. The procedure used here is different from the ones given in [11] and [32]. In this procedure, all the derivatives associated with the grad-div operator have been transferred to the basis/testing function, $\vec{f}$, which is differentiable prior to discretization. Therefore, there are no errors in this procedure corresponding to those introduced by the finite difference approximations used in [11] and [32]. The integral equation is now expressed by the 2-D discrete sequence $I_{x,y}(i,j)$, $\Gamma_{Ax,y}(i,j)$ and $\Gamma_{\Pi}(i,j)$. The discrete sequences, I_x and I_y, represent the sampled values of current density at $\{x = i\Delta x, (j-1)\Delta y \le y \le \Delta y\}$ and $\{(i-1)\Delta x \le x \le \Delta x, y = j\Delta y\}$ for J_x and J_y, respectively. The functions $\Gamma_{Ax}(i,j)$ and $\Gamma_{Ay}(i,j)$ represent the surface integrated magnetic vector Green's functions based on the basis functions applied to the $(i,j)_x$ and $(i,j)_y$ current cells, while $\Gamma_{\Pi}(i,j)$ represents the surface integrated scalar charge Green's function based on the pulse function applied to the (i,j) charge cell. We denote $\Gamma_{Ax}(i,j)$ and $\Gamma_{Ay}(i,j)$ as *the discrete vector Green's functions*, and denote $\Gamma_{\Pi}(i,j)$ as *the discrete scalar Green's function.* These are calculated numerically in the spatial domain before being used in the CG-FFT algorithm. The condition that must first be satisfied, prior to carrying out this operation, is the availability of the spatial Green's functions G_{Ax}, G_{Ay} and G_V.

From the discrete form, we can easily conclude that 2.43 is composed of a linear convolution, such as ($*$: linear convolution)

$$\sum_{i'=1}^{M+1}\sum_{j'=1}^{N+1} I_x(i',j')\Gamma_{\Pi}(i+1-i',j-j') = I_x(i,j) * \Gamma_{\Pi}(i+1,j). \tag{2.47}$$

Thereafter,

$$
\begin{aligned}
LJ_{sx} &= \frac{1}{\Delta x}\left\{\frac{c_1}{\Delta x}[I_x(i,j) - I_x(i-1,j)] + \frac{1}{\Delta x}\frac{c_1}{\Delta y}[I_y(i,j) - I_y(i,j-1)]\right\} \\
&* \quad [\Gamma_\Pi(i+1,j) - \Gamma_\Pi(i,j)] - c_2 I_x(i,j) * \Gamma_{Ax}(i,j). \qquad (2.48)
\end{aligned}
$$

Before we can use the FFT, the linear convolution has to be converted to a circular convolution. Therefore, the discrete sequences $I_x(i,j), I_y(i,j), \Gamma_{Ax}(i,j), \Gamma_{Ay}(i,j)$ and $\Gamma_\Pi(i,j)$ must be extended to periodic sequences. To expand the current sequence, we first define the finite duration sequence, which includes zero padding, i.e.,

$$
I_x^e(i,j) = \begin{cases} I_x(i,j) & 1 \le M; \;\; 1 \le j \le N+1 \\ 0 & M+1 \le i \le 2M+1; \;\; N+2 \le j \le 2N+1 \end{cases} \qquad (2.49)
$$

$$
I_y^e(i,j) = \begin{cases} I_y(i,j) & 1 \le M+1; \;\; 1 \le j \le N \\ 0 & M+2 \le i \le 2M+1; \;\; N+1 \le j \le 2N+1. \end{cases} \qquad (2.50)
$$

The extended current sequences are defined as

$$
\begin{aligned}
\Im_x(m,n) &= I_x^e(m[\text{mod } 2M+1], n\ [\text{mod } 2N+1]), \quad and \qquad (2.51) \\
\Im_y(m,n) &= I_y^e(m[\text{mod } 2M+1], n\ [\text{mod } 2N+1]). \qquad (2.52)
\end{aligned}
$$

For the discrete Green's functions,

$$
\Gamma_\xi^e(m,n) = \begin{cases} \Gamma_\xi(m,n) & 1 \le m \le M+1, 1 \le n \le N+1 \\ \Gamma_\xi(2M+2-m,n) & M+1 \le m \le 2M+1, 1 \le n \le N+1 \\ \Gamma_\xi(m,2N+2-n) & 1 \le m \le M+1, N+1 \le n \le 2N+1 \\ \Gamma_\xi(2M+2-m,2N+2-n) & M+2 \le m \le 2M+1, N+2 \le n \le 2N+1 \end{cases} \qquad (2.53)
$$

$$
\Re_\xi(m,n) = \Gamma_\xi^e(m[\text{mod } 2M+1], n\ [\text{mod } 2N+1]) \qquad (2.54)
$$

where ξ denotes Ax, Ay or Π.

Now 2.48 can be written in a cyclic convolution form($\otimes$: circular convolution) as,

$$\begin{aligned} LJ_{sx} &= \frac{1}{\Delta x}\left\{\frac{c_1}{\Delta x}[\Im_x(i,j)-\Im_x(i-1,j)]+\frac{1}{\Delta x}\frac{c_1}{\Delta y}[\Im_y(i,j)-\Im_y(i,j-1)]\right\} \\ &\otimes \; [\Re_\Pi(i+1,j)-\Re_\Pi(i,j)] - c_2\Im_x(i,j)\otimes\Re_{Ax}(i,j). \end{aligned} \tag{2.55}$$

Applying the convolution theorem, we have

$$\begin{aligned} L\vec{J} = & -\mathcal{F}_D^{-1}\left\{\tfrac{c_1}{\Delta x}[\tfrac{1}{\Delta x}(1-\delta_x^*)\tilde{\Im}_x+\tfrac{1}{\Delta y}(1-\delta_y^*)\tilde{\Im}_y](\delta_x-1)\tilde{\Re}_\Pi+c_2\tilde{\Im}_x\tilde{\Re}_{Ax}\right\}\hat{x} \\ & -\mathcal{F}_D^{-1}\left\{\tfrac{c_1}{\Delta y}[\tfrac{1}{\Delta x}(1-\delta_x^*)\tilde{\Im}_x+\tfrac{1}{\Delta y}(1-\delta_y^*)\tilde{\Im}_y](\delta_y-1)\tilde{\Re}_\Pi+c_2\tilde{\Im}_y\tilde{\Re}_{Ay}\right\}\hat{y} \end{aligned} \tag{2.56}$$

where

$$\delta_x = e^{\frac{j\pi\omega_i}{N+2}} \quad and \quad \delta_y = e^{\frac{j\pi\omega_j}{M+2}}.$$

The functions $\tilde{\Im}_x(\omega_i,\omega_j), \tilde{\Im}_y(\omega_i,\omega_j), \tilde{\Re}_\Pi(\omega_i,\omega_j), \tilde{\Re}_{Ax}(\omega_i,\omega_j)$ and $\tilde{\Re}_{Ay}(\omega_i,\omega_j)$ are the DFTs of their corresponding $\Im_x, \Im_y, \Re_\Pi, \Re_{Ax}$ and $\Re_{Ay}$ quantities. Using the definition of the adjoint operator, i.e., $< I, LJ > = < L^a I, J >$, the quantity $L^a\vec{J}$ can now be derived. It has a form which is similar to that of 2.56.

We have now arrived at the point where the CG algorithm can be used to obtain the solution of the discretized linear system given by 2.42-2.46. From the above description, it is clear that each iteration of the algorithm requires the DFT's of Γ_{Ax}, Γ_{Ay} and Γ_Π. As given in 2.44 and 2.45, Γ_{Ax} and Γ_{Ay} are the integrations of the products of the spatial Green's functions with the basis functions, and Γ_Π is the integration of the product of the spatial Green's function with the derivative of the basis function. They are the key set of quantities used by this algorithm. Therefore, to carry out the spatial discretization procedure, the spatial domain Green's function must be known *a priori.* This poses no problem for free space problems where the spatial Green's function is very simple. But for microstrip structures, this is the major obstacle in our implementation of the spatial discretization scheme. This issue will be addressed more fully in the next chapter. In the following section, we discuss the advantages and disadvantages of this scheme when compared with other common techniques.

2.5 Comparison of Different CG-FFT Schemes

Following the initial introduction of the CG-FFT method as a means of analyzing electromagnetic problems, workers have developed a number of schemes for its implementation. Here we categorize those with distinct features and discuss their advantages and disadvantages.

1. The first of these categories consists of techniques that use the conventional CG-FFT algorithm [17] [25] [27] [28]. Those in this category employ the continuous Fourier transform of the Green's function, as well as the pulse function as the basis function and point matching for discretization of the integral equation. The spatial derivatives are replaced by algebraic factors in the spectral domain. This is the most straightforward of all the known schemes. But, as has been discussed in section 2.3, it suffers from either the requirement for large zero padding or significant errors due to the aliasing of $\tilde{G}$ in the spectral domain. In addition, the direct derivatives on the ill-behaved integral kernels also generate erroneous solutions.

2. The second category [29] [30] [31] consists of those techniques that also employ the continuous Fourier transform of the Green's function and direct algebraic manipulation of the derivatives in the spectral domain, but differ

in that they use subdomain basis functions to expand the continuous current density distributions. These basis functions act as filter functions. The values of the spectral-domain Green's functions are suppressed for large values of frequency. This procedure somewhat reduces the aliasing errors that exist in the first scheme, but does not totally eliminate the problem. The errors due to the treatment of the " $\nabla\nabla\cdot$ " still exists.

3. The third group [26] uses roof-top functions for testing the integral equations. It is therefore an improvement on the above schemes. One of the derivatives is transferred to this testing function by invoking Gauss' theorem, the other is left to operate on the vector potential function which is then expanded in terms of the basis functions. It has been shown that much greater accuracy can be achieved using this procedure. Since the continuous $\tilde{G}$ is still employed, the technique has to use a large zero-padded FFT, thereby reducing the efficiency of the solution.

4. The spatially discrete scheme was introduced in [11] and [32]. This scheme can reduce the size of the zero padding used in the FFT while at the same time eliminating aliasing errors. Significant improvements can be achieved using this scheme. However, the errors introduced by the derivative operation have not yet been resolved. In [11], the pulse function is used to expand the current and point matching is used to test the linear system. The spatial derivatives are approximated using 3-point or 5-point finite differences and calculated via the discrete Fourier transform. This procedure, according to [11], can reduce the errors by smoothing out the effect of discontinuities caused by the pulse function expansion and pointwise collocation. The scheme presented by [32] has better performance than that of [11], because [32] transfers one of the derivatives to the differentiable basis function. The

other derivative, however, is still left to operate on the singular kernel.

5. The latest scheme to be developed is presented in this thesis. A similar scheme is also presented in [21]. This scheme has advantages over all the methods described above. In this thesis, we opt to use the spatially discrete technique. In addition, roof-top functions are used as basis functions and the Galerkin's procedure is used to discretize the system. By using vector identities and Gauss' theorem, both of the spatial derivatives in the $\nabla\nabla\cdot$ have been removed from the singular kernel. One has been moved to the basis function and the other to the test function. Because the roof-top function is differentiable, the two derivatives can be solved analytically before applying the circular convolution theorem. In this way, the size of the zero-padding used in the FFT is reduced to its minimum permissible value (only constrained by the sampling theorem). Both the aliasing error and the errors of the spatial derivatives are eliminated completely. From our point of view, it is unfortunate that the CG algorithm used by [21] is most suitable for applications consisting of Hermitian systems. When applied to non-Hermitian systems, such as problems dealing with electromagnetic radiation and scattering, even for well-behaved system matrices, there are certain excitations for which the iteration may stagnate. As well, the algorithm does not converge monotonically. In fact, convergence is not guaranteed.

In Fig.2.6 are given curves showing the rates of convergence for the different schemes described above. A canonical problem is used for the comparison, consisting of scattering from a square PEC plate. The size of the plate is one wavelength on each side and the angle of incidence is normal to the plate. Table 2.5 gives a comparison of the numbers of iterations required by the five techniques to achieve a prescribed residual error. From Fig.2.6, it is clear that methods 2 and

4 generally converge more slowly than both 3 and the proposed methods. For the first 30 iterations, method 4 appears to have the best rate of convergence. After the first 30 iterations, the proposed method is seen to have the best convergence characteristics. Its superior performance is highlighted by the results obtained using the more demanding error criterion (10^{-3}).

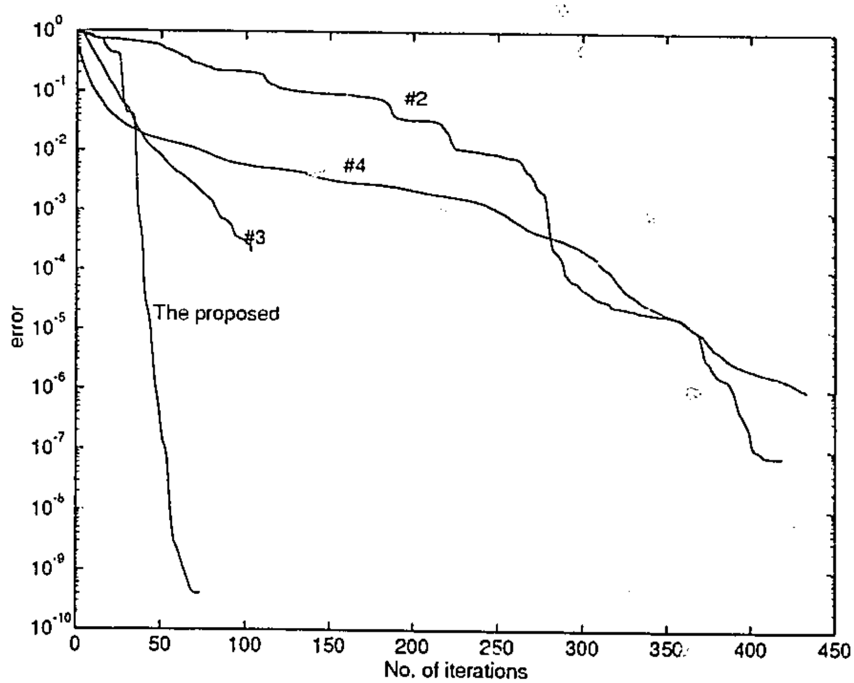


Figure 2.6: Convergence rates for four different CG-FFT schemes. the problem consists of scattering from a $\lambda \times \lambda$ plate where the angle of incidence of the plane wave is normal to the plate.

2.6 Conclusions

In this chapter, we have presented a spatially discrete conjugate gradient - FFT method. This method is found to be very efficient in terms of computer memory

Table 2.1: Comparison of the numbers of iterations required by five different CG-FFT schemes.

Unknown No.	error	App #1	App #2	App #3	App #4	Proposed
17×17	10^{-2}	245	220	47	74	47
17×17	10^{-3}	459	276	70	250	49
37×37	10^{-2}	338	162	128	310	79
37×37	10^{-3}	1480	800	170	1330	92

and computing time. Therefore it is thought to be a candidate for the analysis of large scale electromagnetic problems. Compared with formulations of the CG-FFT which have been used by other workers, the proposed approach has the following advantages:

- The computational domain of the problem under study is discretized in the spatial domain before the convolution theorem is implemented. Thus, the aliasing and truncation errors are eliminated.
- The size of the zero padding used for implementing the FFT is reduced because discretization is applied in the spatial domain.
- Both spatial derivatives of the grad-div operator are removed from the singular kernels and are applied instead to the differential basis and testing functions. The analytical forms of these derivatives are obtained before the discretization. Therefore there is no finite difference approximation involved in this scheme.

Finally, the numerical examples carried out in this chapter have demonstrated that the proposed CG-FFT scheme has advantages over all other published implementations of the CG-FFT.

Chapter 3

The Hybrid CG-FFT Technique

3.1 Introduction

When applying CG-FFT to the analysis of microstrip structures, most researchers use the analytical forms of the Green's functions in the spectral domain because these are readily available. This, inevitably, causes the aliasing problems mentioned above. The spatial window technique, which is equivalent to sampling the Green's function in the spectral domain, is utilized by [33] to reduce the aliasing error. Close approximations of the spatial Green's functions, obtained by directly applying the inverse FFT to a large number of samples of the spectral Green's functions, are utilized in [26] [34]. However, the problems associated with aliasing errors, which have tended to plague CG-FFT, have not yet been satisfactorily solved by the above two schemes. The only way to overcome these difficulties is to first derive the closed-form spatial Green's functions, and then to apply the spatial discretization scheme.

In this thesis, we present a hybrid full-wave analysis technique for microstrip structures, consisting of the spatially discrete CG-FFT method combined with the Complex Discrete Image technique (CDIT). In this hybrid technique, the spectral domain Green's functions are converted into spatial domain analytical Green's functions without losing any full-wave information by using complex image representations. With the closed-form spatial Green's functions at hand, the integral equation describing the microstrip problem in question can be discretized accurately in the spatial domain before the discrete Fourier transform is applied. This can eliminate all of the aliasing or truncation problems that exist in previously published methods, thereby resulting in high efficiency and accuracy. In addition, if one implements the Garlerkin procedure, following the scheme given in chapter 2, the ∇, ∇' operators are moved from the ill behaved Green's functions to the well behaved basis and testing functions. The transposition of the operators also improves the convergence rate and the efficiency.

3.2 CG-FFT Approaches Used for Analyzing Microstrip Structures

It is well-known that the Green's functions which characterize microstrip problems consist of Sommerfeld-type integrals with highly oscillatory and slowly decaying integrands. It is impractical to attempt a direct numerical solution of these integrals. For most applications of CG-FFT, the analytical form of the Green's function in the spectral domain is used to avoid the time-consuming Sommerfeld integrals [12] [13]. But as mentioned above, this results in aliasing problems. Some efforts have been made in the past to deal with these problems. Here we list two of the previously published methods, which have distinct features.

I. Applications of Window Functions In [33], the Green's function is sampled in the spectral domain, which is equivalent to performing a window function over the spatial Green's function and then making the resulting function periodic. There are two major shortcomings to this approach. The first results from the difficulties that arise when attempting to evaluate the dyadic Green's function given in [33],

$$\tilde{G}^F_{\alpha\beta}(\vec{k}^{mn}) = \frac{(2\pi)^2}{T_x T_y}\int_{-S_y}^{S_y}\int_{-S_x}^{S_x}\tilde{G}_{\alpha\beta}(\vec{k}^{mn}-\vec{k})\tilde{Z}(\vec{k})dk_x dk_y.$$

To avoid aliasing in the spatial domain, the window functions Z_x and Z_y must be band-limiting functions, which lead to $\tilde{Z}_x$, $\tilde{Z}_y$ being extended over the complete spectral domain ($S_x, S_y \to \infty$). Because of the integration from $-\infty$ to ∞, it is unrealistic to carry out the integration numerically. Therefore, some truncation must be implemented in the spectral domain. The consequence is that we end up with window functions which are not spatially band-limited. This means that some aliasing errors will be presented, leading to inaccuracies, such as spurious ripples on the current distributions. Different types of window functions can be used, as shown in Fig.3.7, in an attempt to find the optimum one. Following numerical experiments, it is concluded by [33] that the Welch function offers the best performance. But, all of these window functions, except Boxcar(pulse) functions, exhibit characteristics which are diametrically opposed to the known current distributions at the edges of the patches under study. As shown in Fig.3.9, the current distribution on a square conducting patch has a shape which is saddle-like. If one applies a window function along a B-B' cut in the current distribution, he is essentially forcing a weighting function on the current distribution which opposes the natural tendency of the current to peak along the edges. This is another shortcoming of the windowing technique.

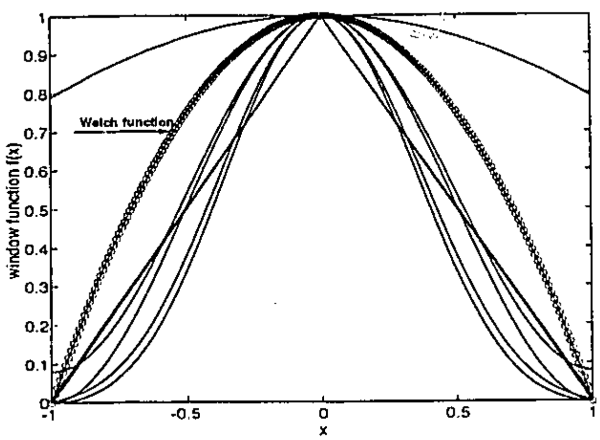


Figure 3.7: Window functions used in [32] to reduce the aliasing errors.

II. The Double FFT Scheme In [26] and [34], a double FFT scheme has been tried, where the analytical spectral Green's functions are sampled in the spectral domain and transferred to the spatial domain using inverse FFTs. The Green's functions are then truncated and transferred back to the spectral domain using zero-padded FFT whose size is twice of that of the truncated Green's functions in each dimension. In this manner, the analytical spectral Green's functions are replaced by approximations which are cyclic in the spectral domain. It is reported that this procedure results in a considerable improvement in the algorithm's rate of convergence. A limitation of this double FFT approach is the requirement, on the part of the practitioner, to keep a large number of data samples because of the unconstrained bandwidth of the spatial Green's function. Otherwise the inherent errors associated with the spectral and spatial truncations would render the solution invalid. An asymptotic extraction technique has been tried by [34]. Unfortunately, this technique did not eliminate the aliasing and truncation errors.

Now it would be expected that if the closed-form spatial Green's function is derived for microstrip structures, the spatially discrete scheme in Chapter 2 can then be employed so as to realize accurate and efficient solutions. However, the problem is how does one overcome the difficulties associated with solving the Sommerfeld integrals.

3.3 Closed-form Spatial Domain Green's Functions

During the last few years, a novel technique, called the complex discrete image technique, has been developed to derive the closed-form Green's functions of microstrip structures from their spectral domain forms. This technique was first introduced in [24] and later was improved by [35] to make it better able to deal with different types of substrates and superstrates.

The Green's functions corresponding to the vector and scalar potentials can be written in the form of a Sommerfeld integral,

$$G_{A,V} = \frac{1}{4\pi} \int_{PSI} k_\rho H_0^2(k_\rho \rho) \tilde{G}_{A,V}(k_\rho) dk_\rho \tag{3.57}$$

where $H_0^{(2)}$ denotes the Hankel function of the second kind, $\tilde{G}_A$ denotes the spectral-domain vector potential Green's function, $\tilde{G}_V$ denotes the spectral-domain scalar potential Green's function and PSI denotes the path of the Sommerfeld integral.

To derive the spatial domain Green's functions starting with their spectral representations, it is necessary to perform the above inverse Hankel transform. Generally, this inverse cannot be solved analytically. But there is a function, consisting of the complex exponentials, which can be analytically transferred to the spatial domain using the Sommerfeld identity,

$$\frac{e^{-jkz}}{r} = \int_{-\infty}^{\infty} \frac{1}{j2k_z} e^{-jk_z z} H_0^{(2)}(k_\rho \rho) k_\rho dk_\rho. \tag{3.58}$$

Mathematically, the idea underlying the CDIT is to use Prony's method to simulate the spectral domain Green's functions by a series of complex exponential functions. The Sommerfeld identity 3.58 is then used to convert the spectral domain Green's functions into a closed form version in the spatial domain. This treatment not only makes it possible to efficiently invert the spectral domain Green's functions, it also provides a formulation for which it is easy to provide a clear physical description.

In the method, a closed-form Green's function for a microstrip structure is divided into three parts

$$G = G_0 + G_{sw} + G_{ci}. \tag{3.59}$$

G_0 represents the contribution from a few quasi-dynamic images dominating in the near-field region. G_{sw} represents the contribution from surface waves dominating in the far-field region of the substrate surface. G_{ci} represents the contribution from the complex images related to leaky waves and which are important in the intermediate region. The procedure used in this technique, that of extracting the three terms of 3.59 from the Hankel transform form of the Green's function, is equivalent to solving the closed-form Green's function. The details of the technique can be found in [24] [35] and results based on this technique are presented here.

3.3.1 The Spectral Domain Green's Functions

The first step of the CDIT technique is the derivation of both the vector and scalar potential spectral domain Green's functions for a specific microstrip structure (composed of single or multilayer substrates). For the microstrip geometry shown in Fig.3.8, the Green's functions of the vector and scalar potentials in the spectral domain are

$$\tilde{G}_A^{x} = \frac{\mu}{4\pi}\frac{1}{j2k_{z0}}\left[e^{-jk_{z0}(z-z')} + R_{TE}e^{-jk_{z0}(z+z')}\right], \quad and \tag{3.60}$$

$$\tilde{G}_V = \frac{1}{4\pi\epsilon_0}\frac{1}{j2k_{z0}}\left[e^{-jk_{z0}(z-z')} + (R_{TE}+R_q)e^{-jk_{z0}(z+z')}\right]. \tag{3.61}$$

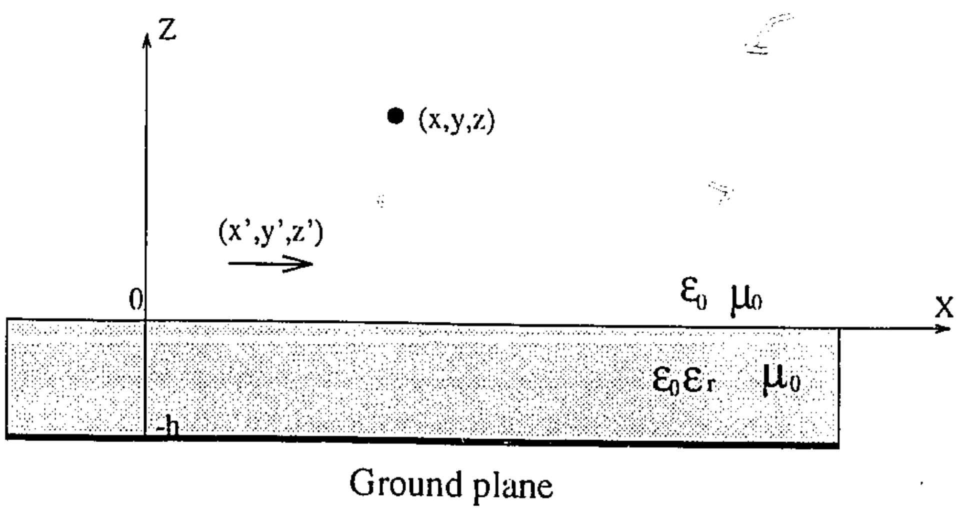


Figure 3.8: A structure of dielectric substrate with ground plane.

In the above equations, $\tilde{G}_A^{x}$ stands for the x component of spectral-domain vector potential $\vec{A}$ created by the x-directed electric dipole ($\tilde{G}_A^{y}$ is of similar form), and $\tilde{G}_V$ stands for the spectral-domain scalar potential associated with one charge on the test dipole. R_{TE} and R_q are the coefficients associated with the TE,TM wave reflections that occur at the interface between the substrate and air. These coefficients describe the effect of the substrate on the performance of the antenna and can be derived using the wave matrix technique [36]. The results are

$$R_{TE} = -\frac{r_{10}^{TE} + e^{-j2k_{z1}h}}{1 + r_{10}^{TE}e^{-j2k_{z1}h}}, \tag{3.62}$$

$$R_q = \frac{2k_{z0}^2(1-\epsilon_r)(1-e^{-j2k_{z1}h})}{(k_{z1}+k_{z0})(k_{z1}+\epsilon_r k_{z0})(1+r_{10}^{TE}e^{-jk_{z1}h})(1-r_{10}^{TM}e^{-j2k_{z1}h})}, \tag{3.63}$$

where

$$r_{10}^{TE} = \frac{k_{z1}-k_{z0}}{k_{z1}+k_{z0}}, \qquad r_{10}^{TM} = \frac{k_{z1}-\epsilon_r k_{z0}}{k_{z1}+\epsilon_r k_{z0}}, \tag{3.64}$$

$$k_{z0}^2 + k_\rho^2 = k_0^2, \quad and \quad k_{z1}^2 + k_\rho^2 = \epsilon_r k_0^2. \tag{3.65}$$

3.3.2 The Quasi-Dynamic Images

Using the Sommerfeld identity, we can find the quasi-dynamic images, which corresponds to the real image, i.e., $\frac{e^{-jk_0 r}}{r}$, and determine their contributions to both the vector and scalar potential Green's functions.

At the low end of the spectrum, it follows that $k_0 \approx 0$ and $k_{z0} \approx k_{z1}$. Therefore, R_{TE} and R_q can be reduced to

$$R_{TE} \leftarrow R_{TE0} = -e^{-jk_{z0}h}, \quad and \tag{3.66}$$

$$R_q \leftarrow R_{q0} = \frac{K(1-e^{-j4k_{z0}h})}{1-Ke^{-jk_{z0}h}}, \tag{3.67}$$

where $K = (1-\epsilon_r)/(1+\epsilon_r)$. Extracting these terms from the integrands in 3.57 and using the Sommerfeld identity 3.58, the quasi-dynamic terms are as follows,

$$G_{A0}^{xx} = \frac{\mu_0}{4\pi}[\Gamma(r_0) - \Gamma(r_2')],$$

$$G_{V0} = \frac{1}{4\pi\epsilon_0}\left[\Gamma(r_0) + K\Gamma(r_0') + \sum_{n=1}^{N} K^{n-1}(K^2-1)\Gamma(r_n')\right], \tag{3.68}$$

$$\Gamma(r) = \frac{e^{-jk_0 r}}{r}, \tag{3.69}$$

$$r_0 = \sqrt{\rho^2 + (z - z')^2}, \quad and \tag{3.70}$$

$$r'_n = \sqrt{\rho^2 + (z + z' + 2nh)^2}. \tag{3.71}$$

3.3.3 The Surface Waves

On the real axis of the complex k_ρ plane , R_{TE}, R_q have poles $k_{\rho p}$. Using the residual theorem, the terms corresponding to surface waves are extracted from 3.57, giving

$$G^{xx}_{A,sw} = \frac{\mu_0}{4\pi}(-2\pi j) Res_1 H_0^{(2)}(k_{\rho p}\rho) k_{\rho p}, \quad and \tag{3.72}$$

$$G_{q,sw} = \frac{1}{4\pi\epsilon_0}(-2\pi j) Res_2 H_0^{(2)}(k_{\rho p}\rho) k_{\rho p}. \tag{3.73}$$

The residues of the integrand at the pole $k_\rho = k_{\rho p}$ are

$$Res_1 = \sum_{p(TE)} \left[\frac{e_{-jk_{z0}(z+z')}}{j2k_{z0}}\right]_{k_\rho = k_{\rho p}} \lim_{k_\rho \to k_{\rho p}} (k_\rho - k_{\rho p}) R_{TE}, \quad and \tag{3.74}$$

$$Res_2 = \sum_{p(TE,TM)} \left[\frac{e_{-jk_{z0}(z+z')}}{j2k_{z0}}\right]_{k_\rho = k_{\rho p}} \lim_{k_\rho \to k_{\rho p}} (k_\rho - k_{\rho p})(R_{TE} + R_q). \tag{3.75}$$

3.3.4 The Complex Images

The terms that remain after the extraction of G_0 and G_{sw} can be simulated by the so-called complex images - images with complex variables. Prony's method and the Sommerfeld identity are used in this step. The results are

$$G^{xx}_{A,ci} = \frac{\mu_0}{4\pi}\sum_{i=1}^{N} a_i^{(1)} \Gamma(r_i^{(1)}),$$

$$G_{q,ci} = \frac{1}{4\pi\epsilon_0}\sum_{i=1}^{N} a_i^{(2)} \Gamma(r_i^{(2)}), \quad and$$

$$r_i^{(k)} = \sqrt{\rho^2 + (z + z' - jb_i^{(k)})^2},$$

where $a_i^{(n)}, b_i^{(n)}$ are the complex coefficients that are derived using Prony's method.

3.4 The Combination of Spatially Discrete CG-FFT with the Complex Discrete Image Technique

The combination of the complex discrete image technique with the discrete CG-FFT method is at the point at which the discrete Green's function is calculated. For a given microstrip problem, the information needed for the implementation of the CDIT is the working frequency f, dielectric constant ϵ_r and the thickness h of the substrate. Then the closed form spatial Green's functions are derived for this microstrip structure, that is, the components of the vector magnetic potential $G_{Ax}(x, yz), G_{Ay}(x, y, z)$ and the scalar potential $G_v(x, y, z)$. With these analytical functions in hand, the integral equation describing the microstrip problem is discretized in the spatial domain using roof-top basis functions and Garlerkin's procedure. The discrete Green's functions,

$$\Gamma_{Ax}(i,j) = \int\int_S f_x^{i,j} \left[\int\int_S f_x^{0,0} G_{Ax}(x - x', y - y', z_0) dS'\right] dS, \quad (3.76)$$

$$\Gamma_{Ay}(i,j) = \int\int_S f_y^{i,j} \left[\int\int_S f_y^{0,0} G_{Ay}(x - x', y - y', z_0) dS'\right] dS, \quad (3.77)$$

$$\Gamma_{\Pi}(i,j) = \int\int_S \Pi^{i,j} \left[\int\int_S \Pi^{0,0} G_V(x - x', y - y', z_0) dS'\right] dS, \quad (3.78)$$

are calculated on each of the small current and charge cells that extend over the microstrip antenna's metallic surface. Here, z_0 denotes the z-coordinate of the interface between the substrate conducting surface and air (in the case shown in Fig.3.8, $z_0 = 0$). The procedure is of such efficiency that the computing time required for generating the discrete sequence is negligible compared to the implementation of the CG-FFT algorithm. Only one set of images need to be found for a given substrate, regardless of the configuration of the microstrip structure above it.

The spatially discrete CG-FFT scheme described in chapter 2 can then be applied in a straightforward fashion to analyzing microstrip problems. This scheme overcomes the difficulties one faces when attempting a numerical evaluation of the Sommerfeld integral in order to obtain the spatial Green's function. At the same time, this technique achieves high efficiency and accuracy because it circumvents the aliasing errors that exist in the other CG-FFT techniques which are used for microstrip analysis.

We now describe the technique that is used here for extracting the quasi-static images. We know that

$$\begin{aligned} G_A^{xx} &= G_{A0}^{xx} + \frac{\mu_0}{4\pi}\int_{-\infty}^{\infty} \frac{1}{j2k_{z0}}(R_{TE} - R_{TE0}) \\ &\cdot\ e^{-jk_{z0}(z+z')} H_0^{(2)}(k_\rho \rho) k_\rho dk_\rho, \quad and \qquad (3.79) \\ G_V &= G_{q0} + \frac{1}{4\pi\epsilon_0}\int_{-\infty}^{\infty} \frac{1}{j2k_{z0}}(R_{TE} + R_q \\ &-\ R_{TE0} - R_{q0}) e^{-jk_{z0}(z+z')} H_0^{(2)}(k_\rho \rho) k_\rho dk_\rho, \qquad (3.80) \end{aligned}$$

where R_{TE0} and R_{q0} are the quasi-static forms of R_{TE} and R_q, respectively, i.e.:

$$\begin{aligned} R_{TE} \leftarrow R_{TE0} &= -e^{-jk_{z0}h}, \quad and \qquad (3.81) \\ R_q \leftarrow R_{q0} &= \frac{K(1-e^{-j4k_{z0}h})}{1-Ke^{-jk_{z0}h}}, \qquad (3.82) \end{aligned}$$

with $K = (1-\epsilon_r)/(1+\epsilon_r)$. While pursuing our objective here of extracting terms, it is to be noted that the term G_{q0} in 3.80 corresponds to R_{q0}, which is a slowly convergent series when ϵ_r is large. Chow, et.al. [35] proposed expanding

R_{q0} in a Taylor series and then truncating the series after the first two leading terms. Unfortunately, this is a poor approximation because neither the K factor nor the exponential term go to zero. The most efficient way to get rid of this quasi-static term is to simulate the term R_{qo} by a summation over a small number of exponential terms, using Prony's method. The remaining part of the spectral domain Green's function can then be easily processed using the complex discrete image technique.

3.5 Accuracy and Convergence Considerations for the Hybrid CG-FFT

The problem of calculating the scattering from a rectangular microstrip antenna is used as an example for quantifying rates of convergence. The patch is illuminated by a linearly-polarized plane wave ($f = 3.12GHz$), whose plane of polarization is parallel to the x-axis. The co-polarization and cross-polarization current distributions are shown in Fig.3.9. The co-polar components of the induced current taken along the $A - A'$ and $B - B'$ cuts are compared with the results obtained by the Method of Moments in Fig.3.10. Excellent agreement is obtained. A comparison of the rates of convergence is made between the results given in this thesis and those in [33]. The relative error is defined as

$$E_r = 10 \log \frac{||LJ - E_d||}{||E_d||}.$$

From Fig.3.11, it is clear that the method proposed here has a much better rate of convergence than that in [33], especially when the number of unknowns is large.

To demonstrate the accuracy of the proposed method, a line feed microstrip patch antenna is analyzed. It is known that the input impedance is one parameter

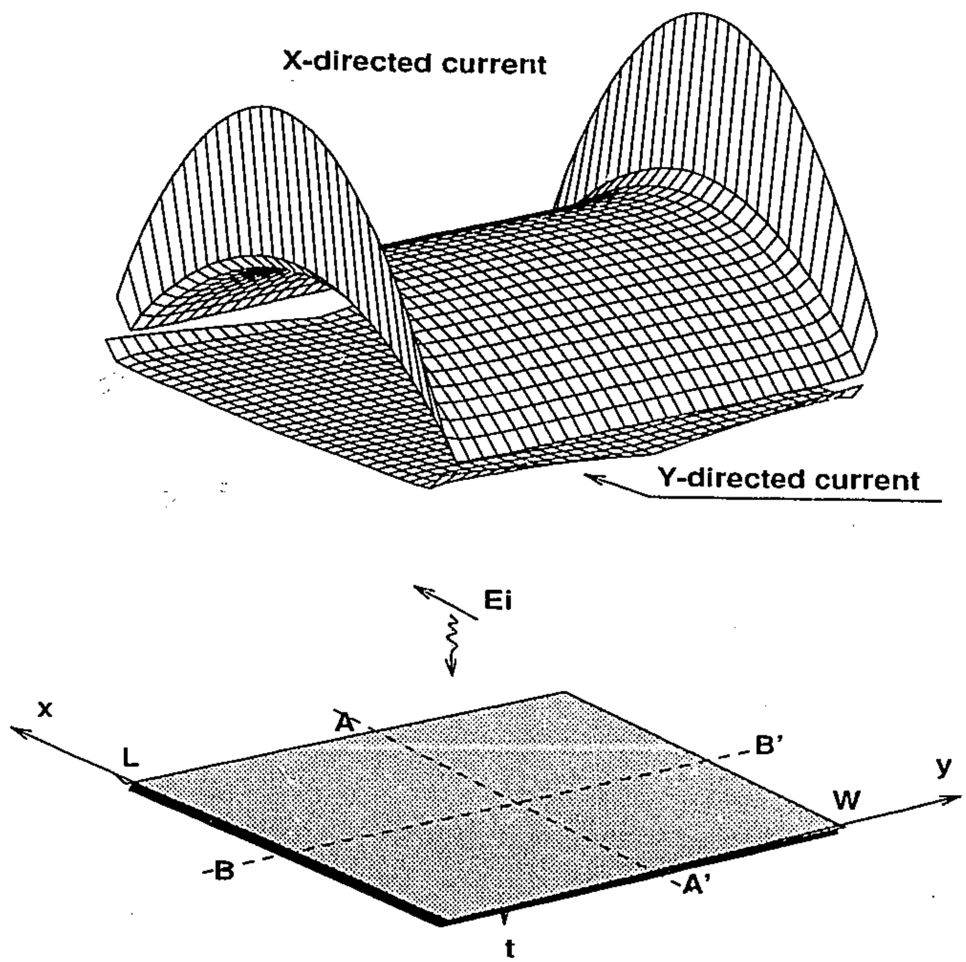


Figure 3.9: Current distributions on a rectangular patch antenna illuminated by x-directed electrical field with frequency of 3.12GHz. $W = 15mm, L = 10mm, \epsilon_r = 10.2, t = 1.27mm$.

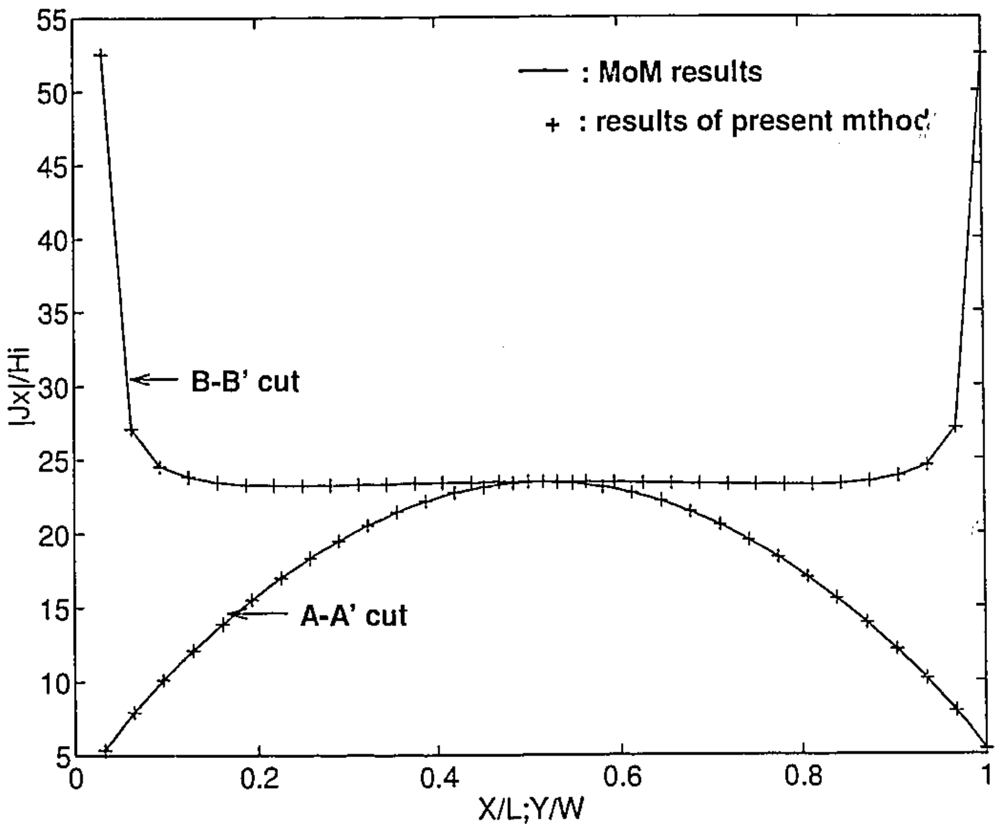


Figure 3.10: Co-polar current (x-directed) along $A - A', B - B'$ cuts of the microstrip patch, compared with the calculated results obtained by the Method of Moments

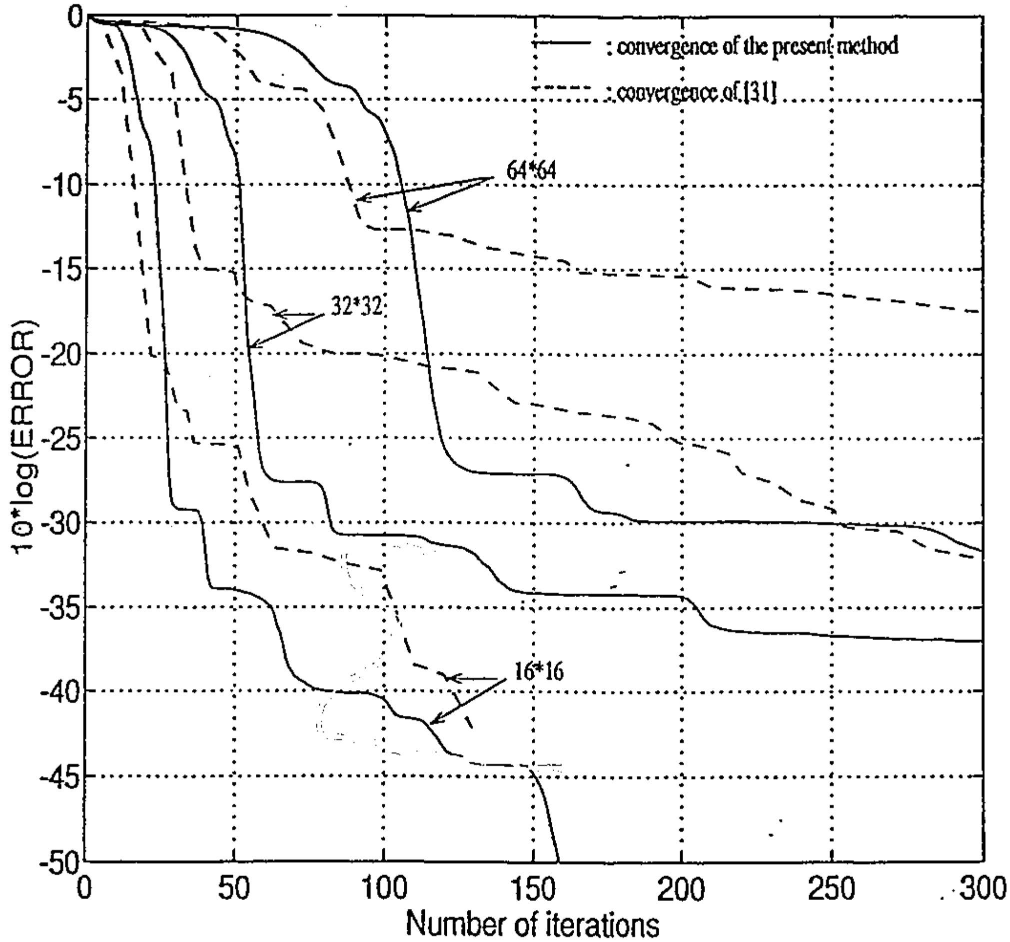


Figure 3.11: The comparison of the rates of convergence for the hybrid CG-FFT technique and the technique used in [32]. The benchmark problem is the microstrip patch shown in Fig.3.9.

that is highly sensitive to inaccuracies in analytical techniques. Therefore, it serves as a good metric for testing the accuracy of our analytical technique. Our values of input impedance are derived from the standing wave patterns on the transmission structures (Fig. 3.12). The results over the frequency range $4GHz - 7GHz$ are shown plotted on a Smith chart in Fig. 3.13. They are compared with measured data as well as results obtained using the FDTD technique [37]. The results of a comparison between measured and simulated results shows that our technique has very good accuracy.

3.6 Conclusions

A full-wave analytical technique for large microstrip structure problems has been presented. It combines the CG-FFT method with the complex discrete image technique. The proposed method has a number of merits, such as;

- The closed-form Green's functions are obtained using the complex image technique without losing any full-wave information and with a negligible computational overhead. The cumbersome numerical evaluation of the Sommerfeld integral is avoided.

- The accurate and efficient spatially discrete CG-FFT technique described in Chapter 2 is incorporated into this algorithm. All of the advantages enjoyed by the technique when applied to free-space problems are transferred to microstrip problems.

- With the incorporation of the results given in Chapter 2, dealing with the derivatives of the div-grad operators, this approach is expected to achieve

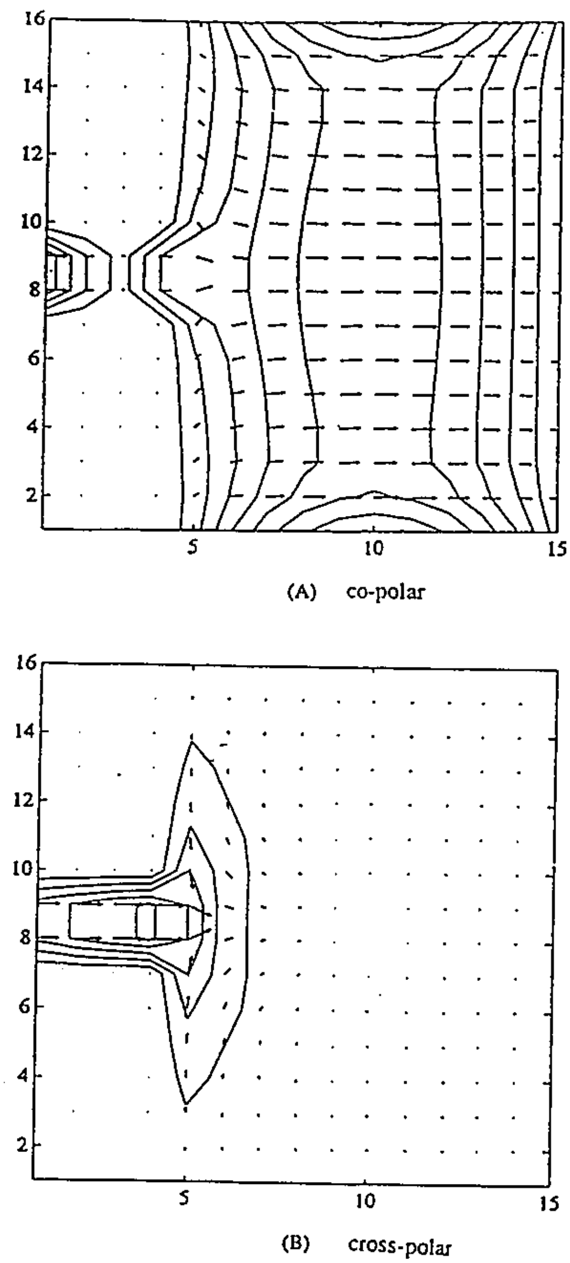


Figure 3.12: The current distribution on a microstrip patch antenna and its feed line.

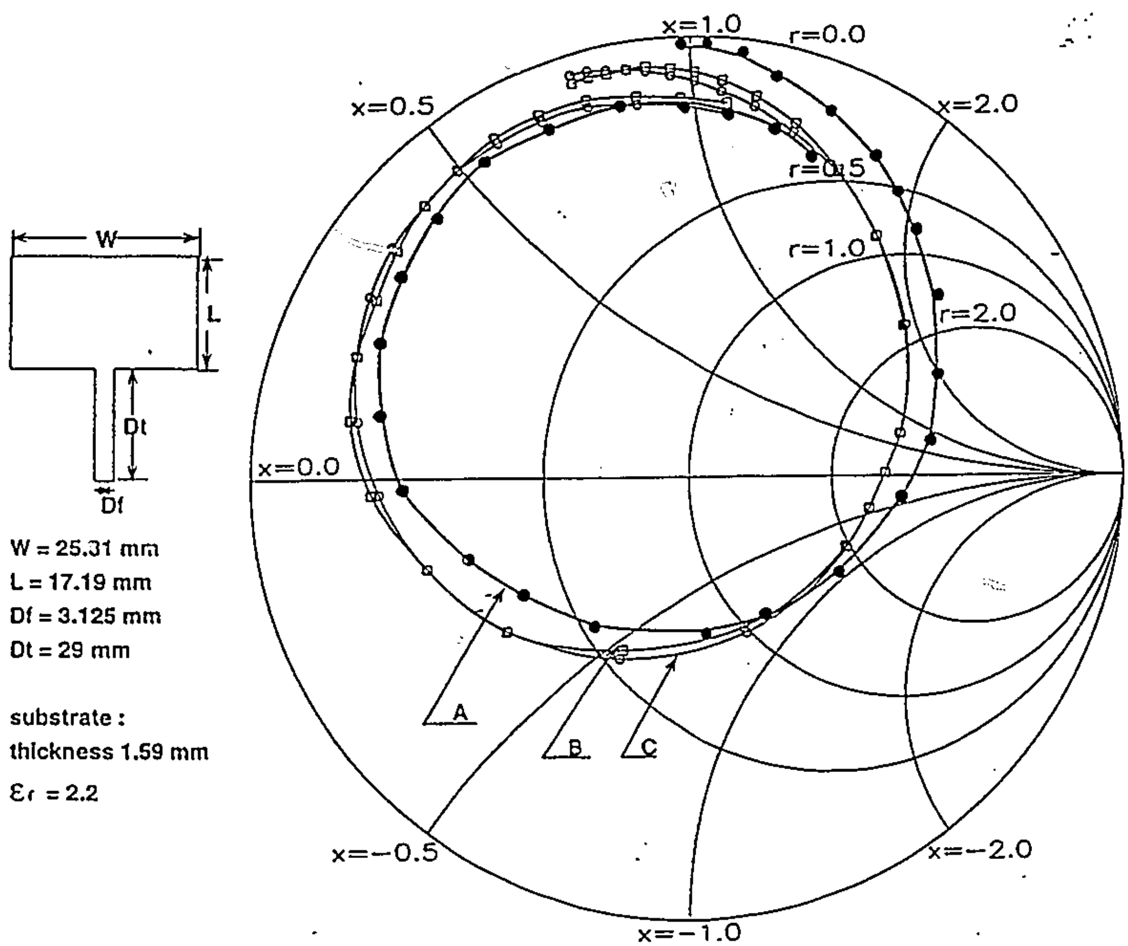


Figure 3.13: The input impedance of a rectangular microstrip patch antenna (frequency : 4 - 7 GHz clockwise). (A): results obtained by the presented method; (B) FDTD results; (C) experiment results.

greater efficiency and to yield more accurate results than other CG-FFT techniques.

It is concluded that this method is very suitable for the analysis of very large microstrip structures.

Chapter 4

The Analysis of Microstrip Antenna Arrays

4.1 Introduction

Due to the complexity which is inherent in the design of microstrip antenna arrays, it is important that efficient analytical techniques be developed. These techniques can be used as computational "engines" in CAD tools. There have been a great number of papers published on the topic of modeling single microstrip elements [38]- [43]. Some papers have put forward techniques for analyzing arrays based on the assumption that these arrays consist of an infinite number of elements. But the problem of modeling finite sized arrays is drawing the attention of workers because there is a requirement for increased accuracy in the results obtained, as well as requirement for increased robustness in the algorithm themselves. [49]-[53].

At the elementary level, the pattern of a single microstrip element can be combined with the array factor for an array of independent point sources to predict the radiation pattern of a microstrip antenna array. Since surface waves can propagate in the dielectric layer that connects the array elements, mutual coupling effects for microstrip arrays may be considerably larger than for other types of array antennas. Therefore, this result will be of limited value if mutual coupling is not taken into account.

The infinite array analysis method is based on the assumption that the array consists of uniform elements and the antenna is of infinite extent with the elements forming a regular grid pattern. A Floquet-type representation of the field from a unit element can then be used to express the field of an array composed of these infinite number of elements. Although this model includes mutual coupling, it cannot be used to treat an array with the complicated configurations shown in Figs.4.37 and 4.56. Another major disadvantage of this approach is that it can neither deal with the array edge effects nor with the effects of spurious radiation from the feed networks. Both of these effects must be taken into account in order to realize good impedance matching and pattern synthesis. It is felt that for future applications, techniques that are based on the simplifying assumptions described above will not provide data that will meet increasingly demanding performance specifications.

Finite large array modeling methods, such as the one proposed here, do not suffer from the above assumptions. They directly take into account all of the electrical characteristics of the array, including the complete feed network. These models will yield results that surpass the accuracy of those obtained using other techniques. But there is a big obstacle that must be overcome when applying these techniques. In finite array analysis, the interaction between each pair of discretized small cells within an array are calculated. This procedure results in a very dense system matrix. When the array under investigation is large, the storage and processing (inversion) of this matrix will require a very large computer memory, as well as very long computing times. From a computational point of view, the implementation of this model is prohibitively difficult.

It is now known that the CG-FFT method can reduce the computational burden required for solving microstrip antenna problems because it requires less computer memory and computing time than other full wave techniques. Combined with the complex discrete image technique, the hybrid CG-FFT method becomes highly suitable for carrying out rigorous full-wave analyses of finite large microstrip antenna arrays [23] [54]. In this chapter, the proposed technique is exploited by carrying out simulations of several different types of the microstrip antenna arrays. First, we carry out simulations of microstrip dipole arrays with different sizes and array configurations. Using the results of the simulations, we are able to study array edge effects; i.e. that fact that the array does not extend infinitely into space, but rather it is truncated, thereby giving boundaries with different shapes. Series fed and corporate fed microstrip planar arrays are then analyzed. A comparison is made between the radiation from the feed structures with that from the radiating elements. In order to demonstrate the full potential of this technique we undertake a very challenging problem, that of simulating a large microstrip reflectarray. This chapter ends with conclusions and a discussion of results.

4.2 Analysis of Planar Microstrip Dipole Arrays

As mentioned in the introduction, the characteristics of large arrays, such as their input impedances and array radiation patterns, are usually derived using the infinite array assumption. However, actual arrays consist of a finite number of elements. In this section, it will be shown that characteristics of elements in a finite arrays differ from those of elements in an infinite array, especially near the edge of the array. The shape of an array can also affect the characteristics of the array; in particular, the radiation pattern.

Edge effects of planar dipole arrays have been investigated principally by comparing calculated or measured data for a finite array with theoretical data which were obtained using the infinite array approximation [55] [56]. Most workers, however, have restricted their analysis to the central elements of large arrays, whose element geometries consist of rectangular grids, or to smaller arrays. This is simply because of the computer limitations for the algorithms they used. Pozar[50] treats the problem of simulating finite-sized arrays of printed dipoles. The effect of array size on the generation of surface waves was investigated. Only rectangular arrays with rectangular grids of elements are treated in [50]. One piecewise sinusoidal basis function per dipole is used in most of the calculations because the authors use the conventional the Method of Moments in their analysis. The method forces them to restrict the number of unknowns they can deal with.

In this section, various types of large planar microstrip dipole arrays are studied. These include arrays with rectangular and hexagonal outer boundaries, and interior geometries consisting of rectangular and triangular grids. With the objective of increasing the accuracy of the results, especially for the case of array input impedance which is particularly sensitive to the value of the current at the excitation point, 9 basis functions are used for each dipole.

The geometry for a finite array of microstrip dipoles is shown in Fig. 4.14. Each dipole is assumed to have length L, width W, and to be fabricated on a

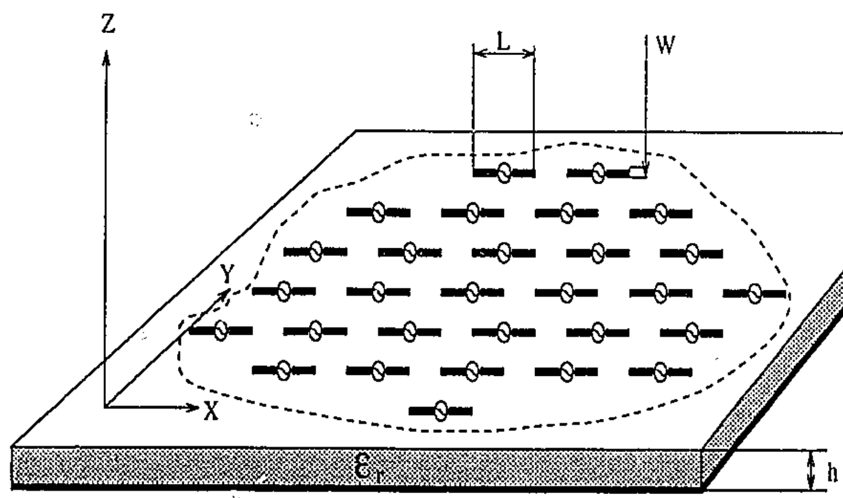


Figure 4.14: Geometry of an arbitrary shaped microstrip dipole array.

substrate with dielectric constant ϵ_r and thickness h. The elements are arranged periodically either in a rectangular grid with spacing a, b in x, y direction, respectively, or in an equilateral triangular grid with element spacing t (see Fig. 4.15). The dipoles are fed at their centers with infinitesimal generators V_T. The equivalent circuit of one of the dipoles is shown in Fig. 4.16.

Since $W \ll \lambda$, the magnitude of the y-directed current is much smaller than that of the x-directed current and can be considered to be negligible. Furthermore to a first approximation, the x-directed current can be considered to be evenly distributed in the y-direction. To solve for this current distribution, an EFIE is

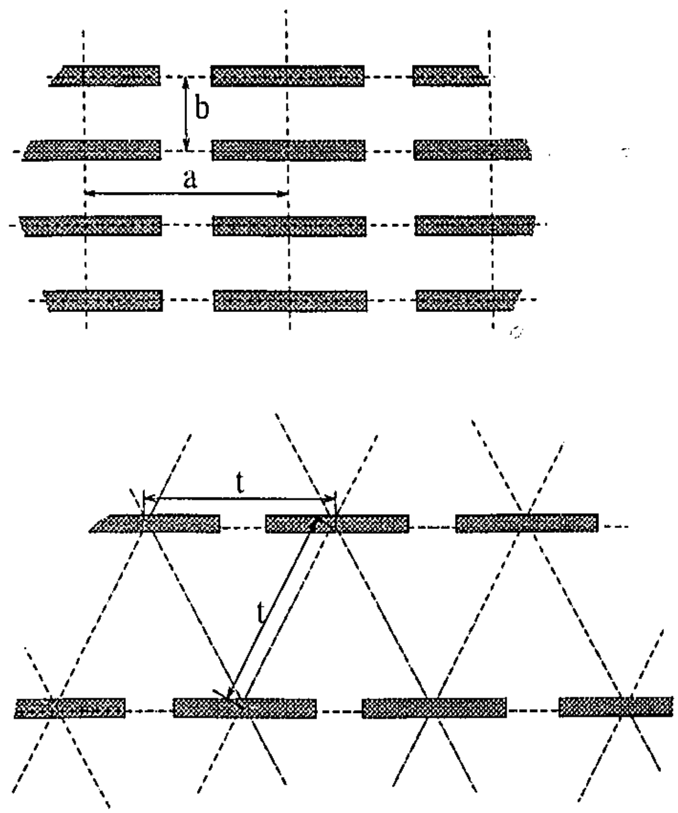


Figure 4.15: Rectangular and triangular grid geometries used in the microstrip dipole arrays.

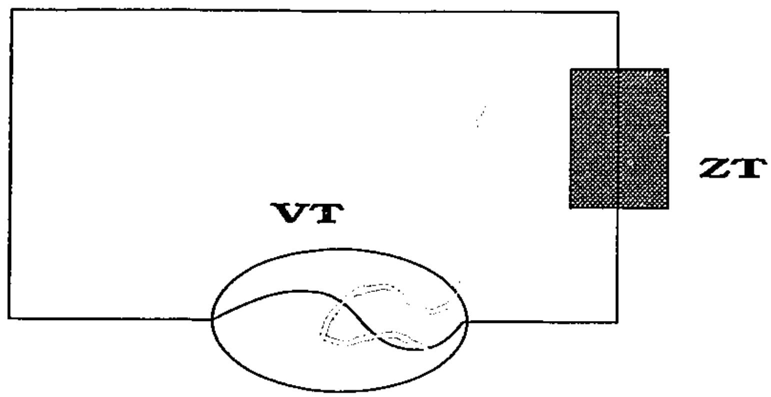


Figure 4.16: The equivalent circuit of a microstrip dipole element.

formed and discretized using the procedure given in Chapter 2, thereby leading to the following linear equation

$$ZJ = V - V_T \qquad \Longrightarrow \qquad ZJ + V_T = V \tag{4.83}$$

where Z, I and V are the impedance matrix, current matrix and excitation voltage matrix, respectively. Here, V_T is the voltage matrix which gives the voltage drop caused by the generator's terminating impedance. If the generators are located at grid points given by (m, n), the elements of V_T are

$$V_T(i,j) = \begin{cases} J(i,j)Z_T(i,j) & \text{if } i = m, j = n \\ 0 & \text{otherwise.} \end{cases} \tag{4.84}$$

When applying the CG-FFT algorithm, the term ZJ can be calculated by modifying 2.56 as follows.

$$ZJ = L\vec{J} = \; -\mathcal{F}_D^{-1}\left\{\frac{c_1}{\Delta x}^2(1-\delta^*)\Im(\delta-1)\tilde{\Re}_\Pi + c_2\Im\tilde{\Re}_A\right\} \tag{4.85}$$

where

$$\delta = e^{\frac{j\pi\omega_i}{N+2}}.$$

$\tilde{\Im}(\omega_i, \omega_j)$, $\tilde{\Re}_{\Pi}(\omega_i, \omega_j)$, and $\tilde{\Re}_A(\omega_i, \omega_j)$ are the 2D DFTs of their corresponding quantities which are defined in Chapter 2. After obtaining the values of ZJ, the value of the RHS of the second form of 4.83 can be evaluated simply by subtracting of V_T from ZJ. Then the CG-FFT procedure can be carried out in a straightforward fashion.

Using the procedure described above, three different configurations are analyzed. In each case, $\epsilon_r = 2.1, h = 1.59mm$. The array geometries are shown in Fig. 4.17. To begin, we are interested in determining how the distribution of current on the array elements changes with the array parameters. In Fig. 4.18 are given the magnitudes of the currents on the rows of a type #1 hexagonal array with $M = 30, N_1 = 10$ and $N = 70$. The total number of the elements in this array is 1166. Fig. 4.19 gives a comparison of the current distributions along the central rows of a type #2 hexagonal array with $M = 47, M_1 = 37, N = 17$ and a rectangular array with $M = 47, N = 17$. The total numbers of the elements in these two arrays are 709 and 799, respectively. It should be noted that the spaces between the elements are omitted in the graphs. It is observed from the above two graphs that the current is more uniform in the central portions of the arrays and that it varies remarkably near the edges of the arrays. It is also found that the current distribution changes with array geometry. For purposes of comparison, an ordinary dipole array (in free space) is simulated. It is a type #2 hexagonal array with $M_1 = 8, M = 15$, and $N = 15$. The width of and the space between the elements of this array are the same as the width and the space of the elements of microstrip dipole array used in the former example. In Figs.4.20 and 4.21 are shown the magnitudes and phase distributions of the currents on the 1st, 4th, 8th, 12th and 15th rows of the array. It is seen that even though variations still exist in the magnitude of the current across the array, the magnitude of the variation is

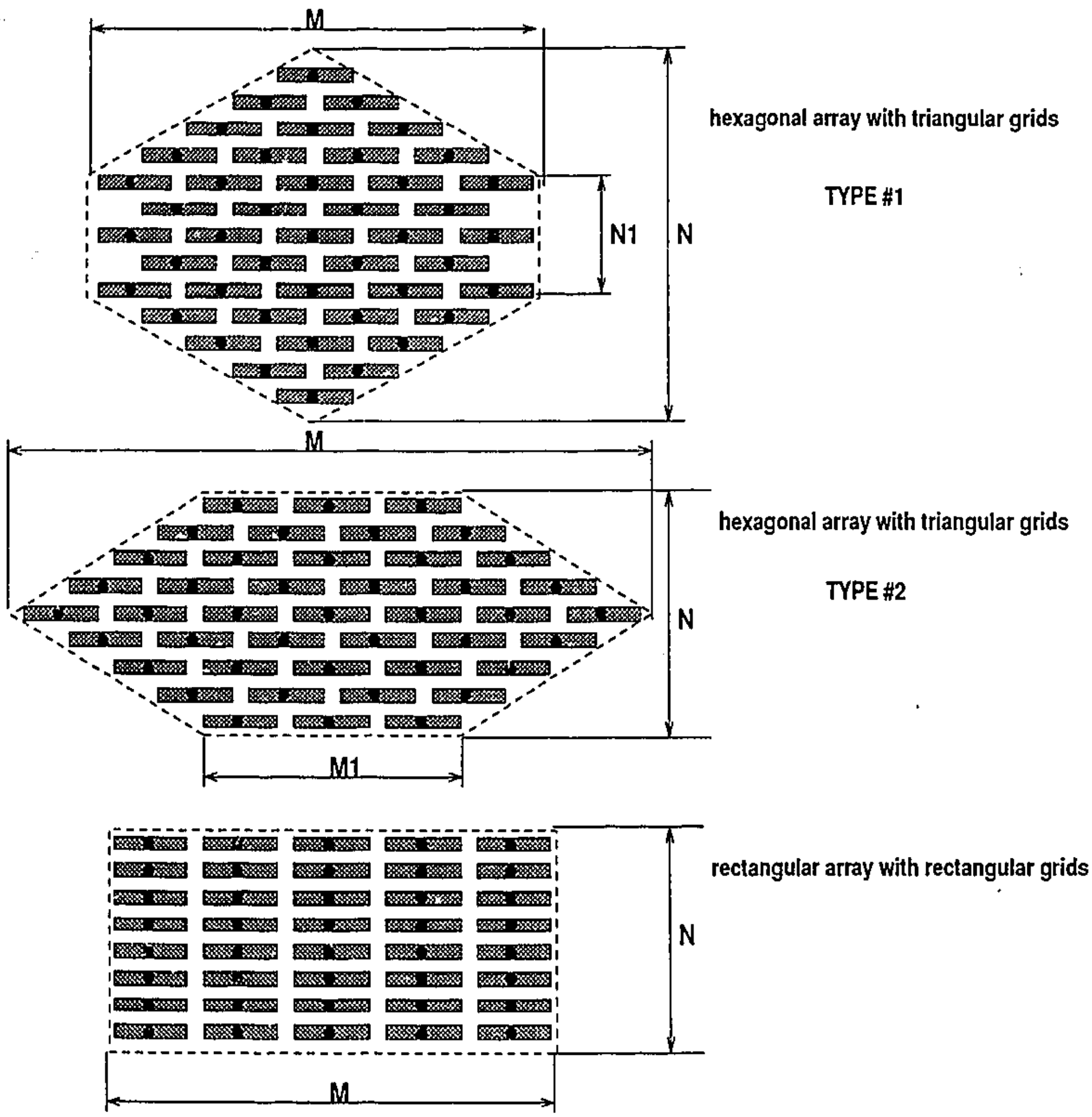


Figure 4.17: Microstrip dipole arrays with three different geometries.

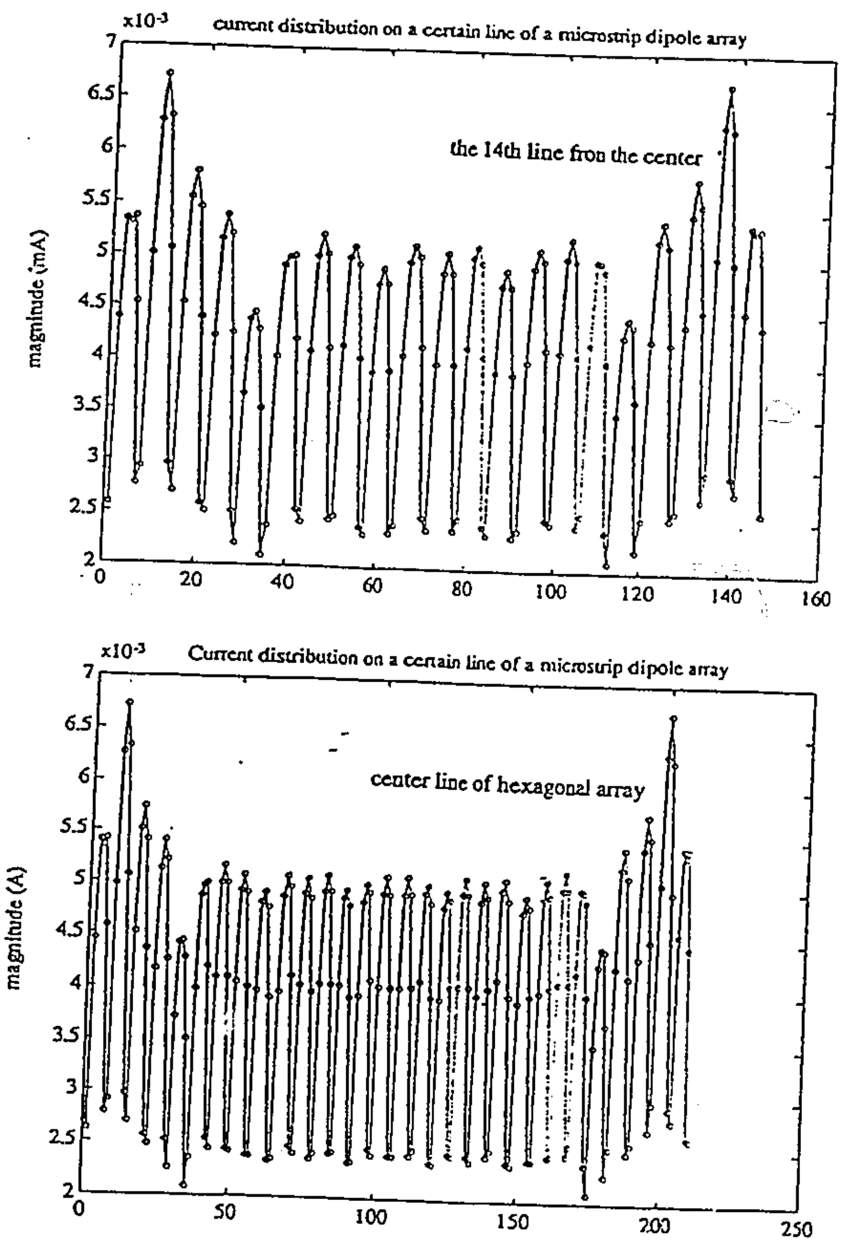


Figure 4.18: The current distributions on different rows of a type #1 hexagonal array with $M = 30, N_1 = 10, N = 70$.

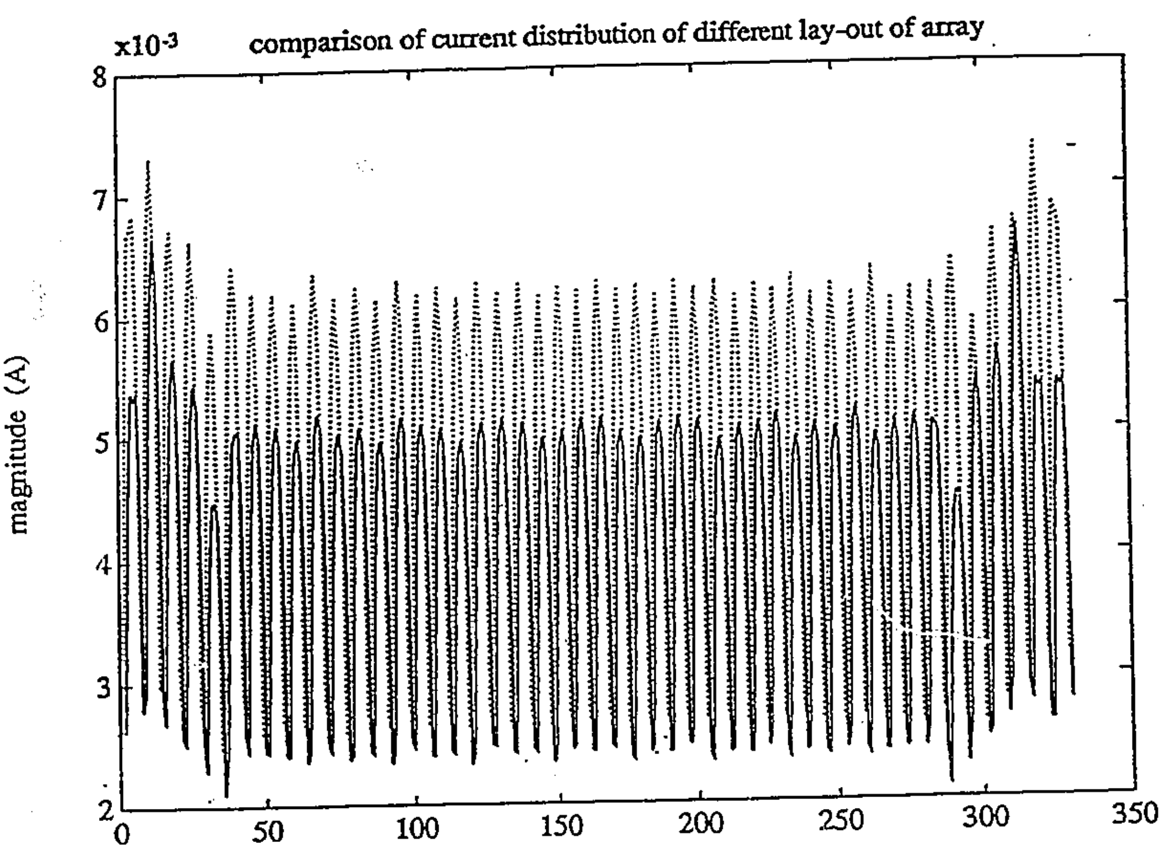


Figure 4.19: The current distributions on the central rows of a type #2 hexagonal array with $M = 30, N_1 = 10, N = 70$ (solid line) and a rectangular array with $M = 47, N = 17$ (dash line).

considerably less than that of one of the microstrip dipole array shown in Fig. 4.19. The differences in the characteristics of these two types of arrays are attributed to the existence or nonexistence of surface waves. In the case of the dipole array in free space, surface waves are virtually nonexistent, whereas for the case of microstrip arrays, surface waves exist and propagate within the substrate. Since surface waves can increase the mutual coupling between elements of an array, it follows that the effects of mutual coupling are likely to be greater for a microstrip array as compared to a dipole array in free space.

The input impedance of the individual elements in an array is also affected by the size of the array. In Fig. 4.22 is shown the input impedance of the central element of $N \times N$ array, where $N = 1, 3, 7, 19, \infty$. In each instance, the array shape is rectangular and the element geometry consists of rectangular grids. The results are compared with those in [47]. The difference between these two data sets may be attributed to the different numbers of basis functions used in the simulations and the different approaches for treating the Green's functions. These data are for the central elements only, but the impedances are expected to change from element to element in a finite array. Figs.4.23 and 4.24 depict the resistance and the reactance of each element in a hexagonal array type #1 with $M = 8, N = 22, N_1 = 8$, presented in the form of contours and gray-scales where denser colors represent larger values of the resistance of the element at that location. However, it should be noted that the contours and the gray-scales are meaningful only at element positions. The results clearly show an overall spatial variation of the input impedance within the array. It is predictable that the impedance of the central element ($82.6 + j7.24\Omega$) is close to the value of that of an infinite array. It is observed that the differences between the impedances of the finite array elements with those of a infinite array are largest at the edge of the arrays and decreases

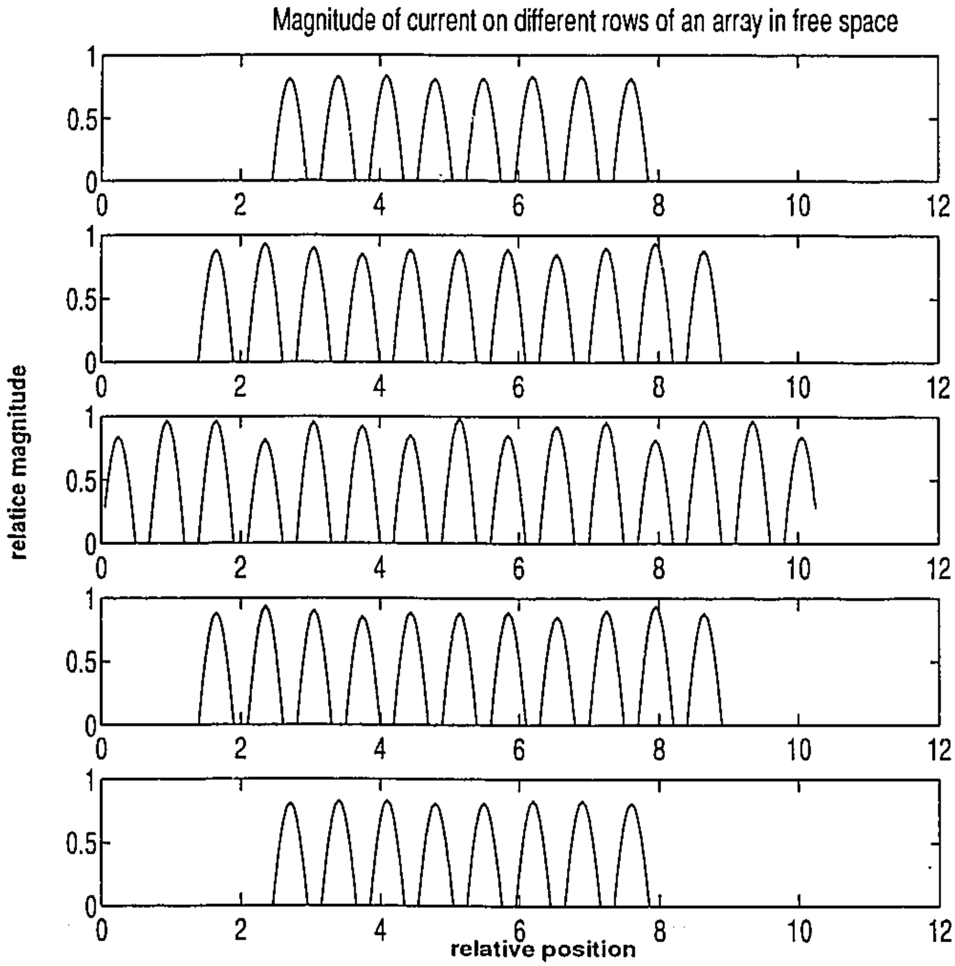


Figure 4.20: The magnitude of the current distribution on the 1st, 4th, 8th, 12th and 15th rows of an ordinary dipole array in free space. The array is a hexagonal array (type #2, $M_1 = 8, M = 15, N = 15$)

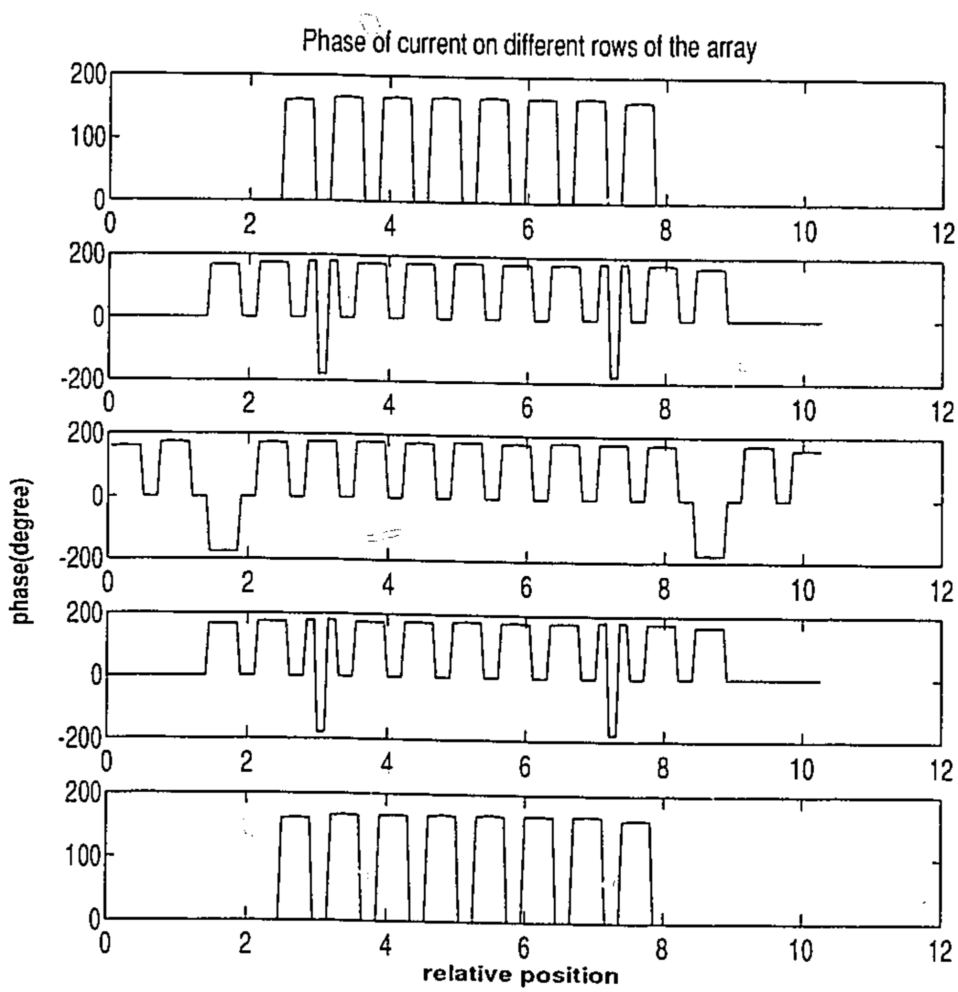


Figure 4.21: The phase of the current distribution on the 1st, 4th, 8th, 12th and 15th rows of an ordinary dipole array given in the previous figure.

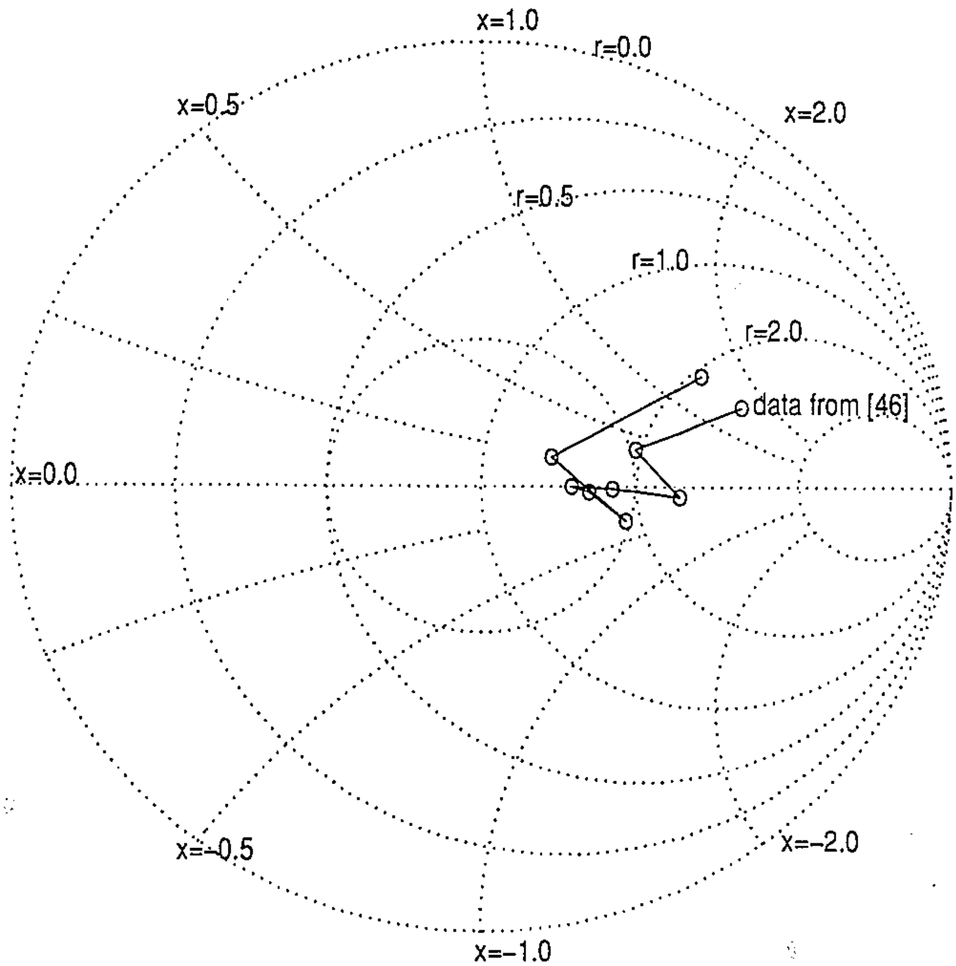


Figure 4.22: The input impedance of the center element of $N \times N$, $N = 1, 3, 7, 19, \infty$ microstrip dipole array with rectangular shape and rectangular grids. The values of pertinent array parameters are: $\epsilon_r = 2.55, h = 0.19\lambda_0, a = b = 0.5\lambda_0, L = 0.39\lambda_0, W = 0.01\lambda_0$.

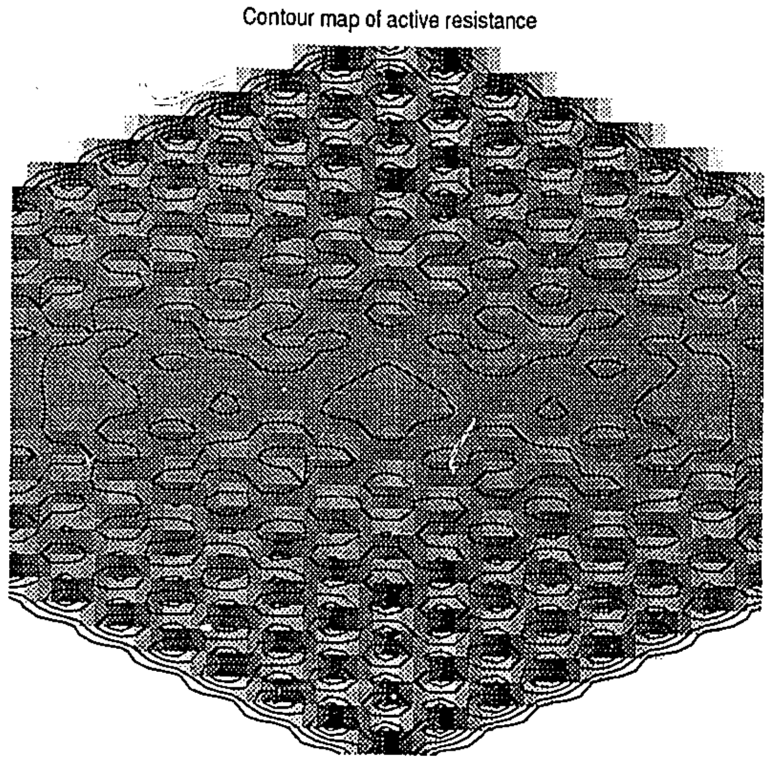


Figure 4.23: The spatial variation of input resistance of a hexagonal microstrip dipole array type #1 with $M = 8, N = 22, N_1 = 8$, substrate $\epsilon_r = 2.2, h = 1.59mm$. The variation is presented by contours and a gray-scale where denser colors represent larger values of the resistance of the element at that location.

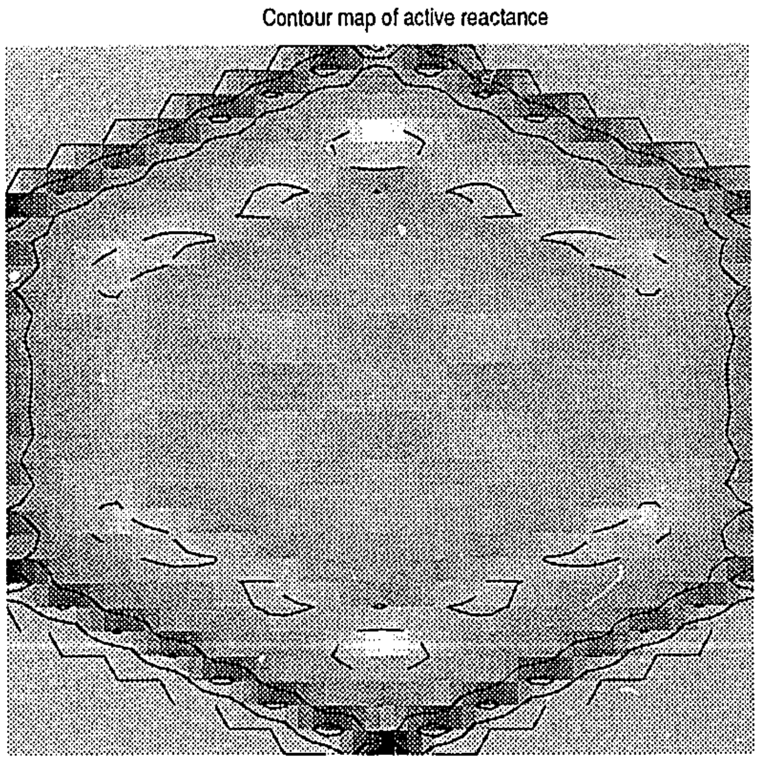


Figure 4.24: The spatial variation of input reactance of the hexagonal microstrip dipole array given in the previous figure. The variation is presented by contours and a gray-scale where denser colors represent larger values of the resistance of the element at that location.

when one approaches the centers. It is to be noted that the input impedances of the elements at the edge are between the value of the infinite array ($82 + j0\Omega$) and that of the single isolated element ($144 + j77.9\Omega$).

In addition to the effect of the array size, edge and shape to the individual elements, these factors also have effects on the overall performance such as the radiation pattern. Fig.4.25 shows the radiation patterns of two arrays with rectangular and hexagonal shapes, respectively (see Fig.4.17). The differences in the radiation patterns of the two arrays are obvious, however, the patterns cannot be simulated if one assumes infinite array models.

4.3 Modeling Planar Microstrip Patch Arrays

The proposed hybrid CG-FFT method can accommodate arbitrarily shaped microstrip planar structures. In this section, planar microstrip patch arrays, such as series-fed arrays and corporate-fed arrays, are analyzed along with their feed networks.

The feed network's effect on the performance of an array has been studied by some authors [53] [57] [58] [59]. In [59], the radiation from and losses due to feed network's are studied by using a surface current approach in which the electrical current on the feed line is modeled as an ideal transmission line. In [57], the effect of radiation on the array pattern is analyzed using a source distribution of magnetic currents. These currents are obtained from a patch cavity model. Both of these models are based on some asymptotic assumptions, which are not full-wave analysis approaches. In [53], a full-wave analysis method is presented for predicting the feed network effects. Both the Moment Method and the discrete

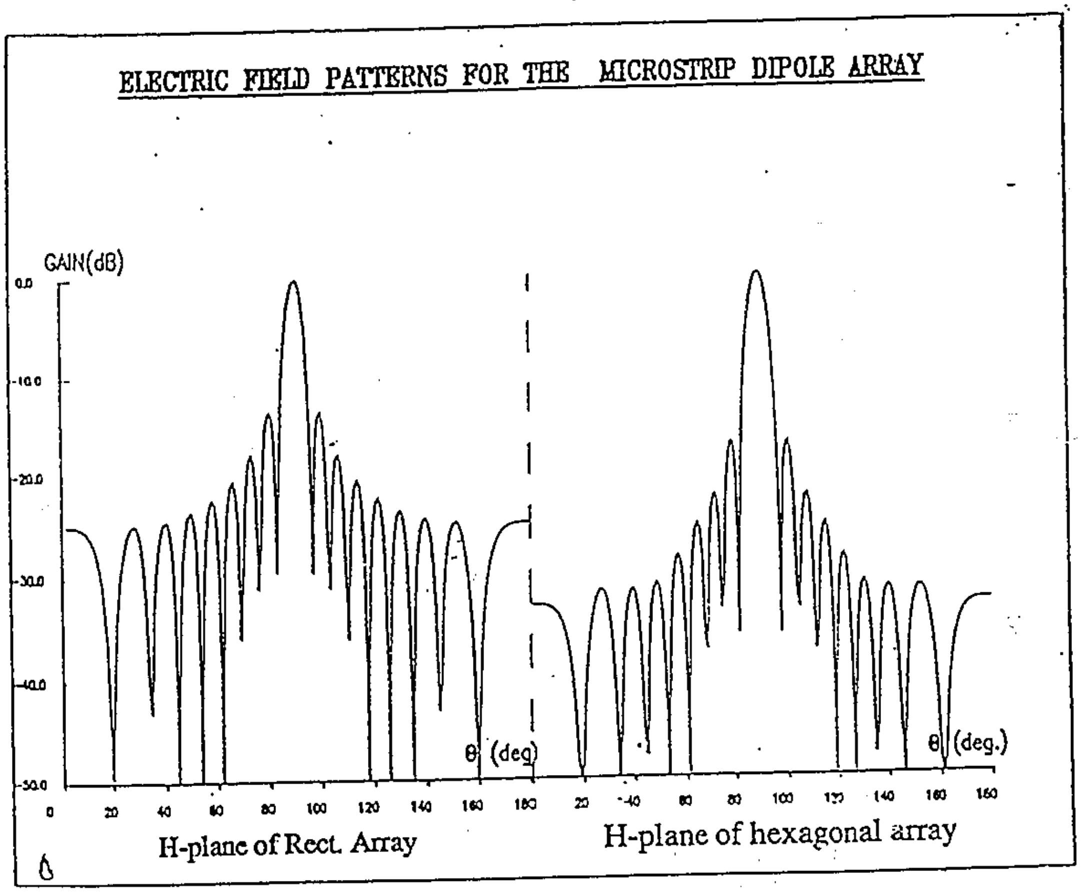


Figure 4.25: The radiation patterns of a microstrip hexagonal array and rectangular array given in Fig.4.11.

images technique are used in this model. The size and the type of the arrays treated in [53] can be classified as being both small and simple. With the aid of the hybrid CG-FFT, this approach can be broadened to cover a much broader range of microstrip antenna arrays.

First, a series-fed microstrip array is simulated. It is an 1×4 patch array with different values of L (shown in Fig.4.26). The array is driven by a δ-gap source located near the end of the feed network.

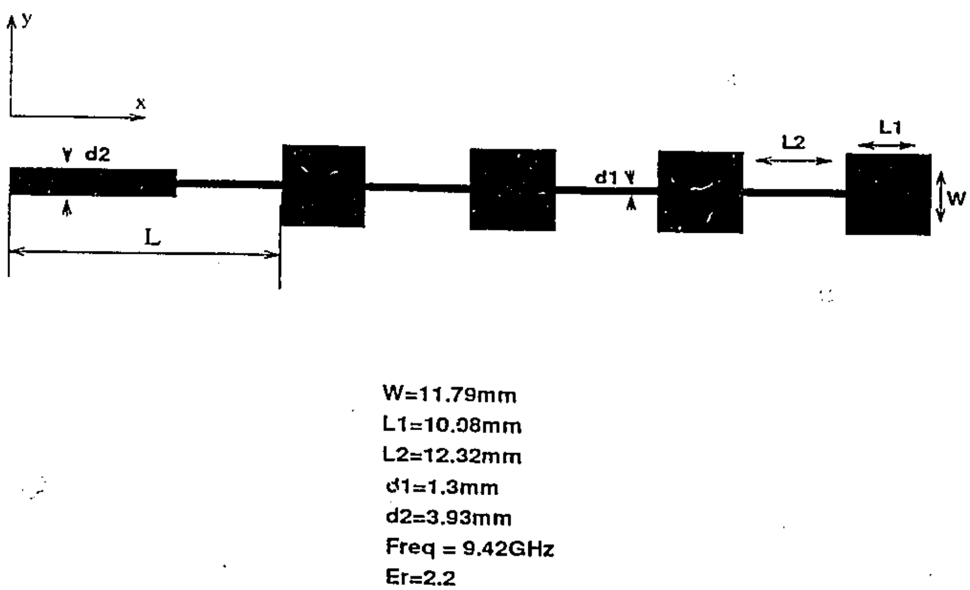


Figure 4.26: A series-fed microstrip array.

Fig.4.27 depicts three-dimensional views of the x- and y-directed currents on the patch elements and the connected feed networks of the array with $L = 37mm$. In Fig.4.28 are shown the current distributions on the central lines of the arrays with three different lengths of L.

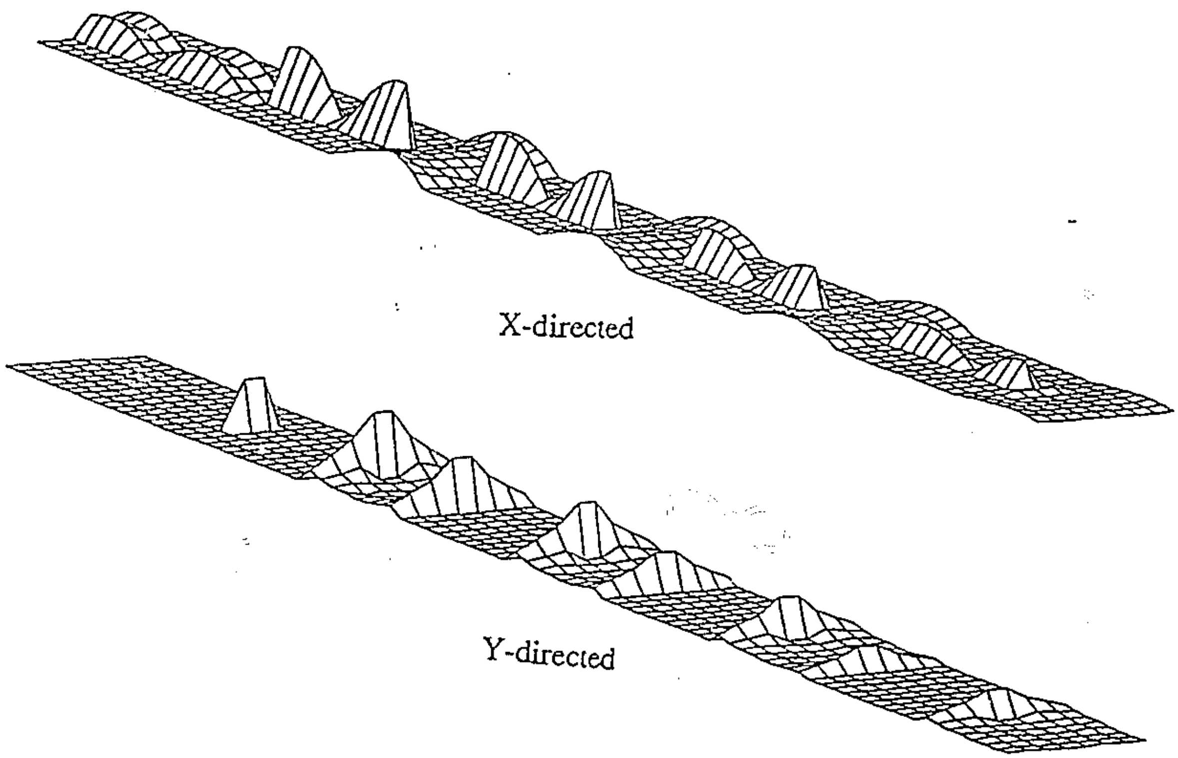


Figure 4.27: The magnitude of x-directed (upper) and y-directed (lower) current distribution on the series-fed microstrip array shown in the precious figure.

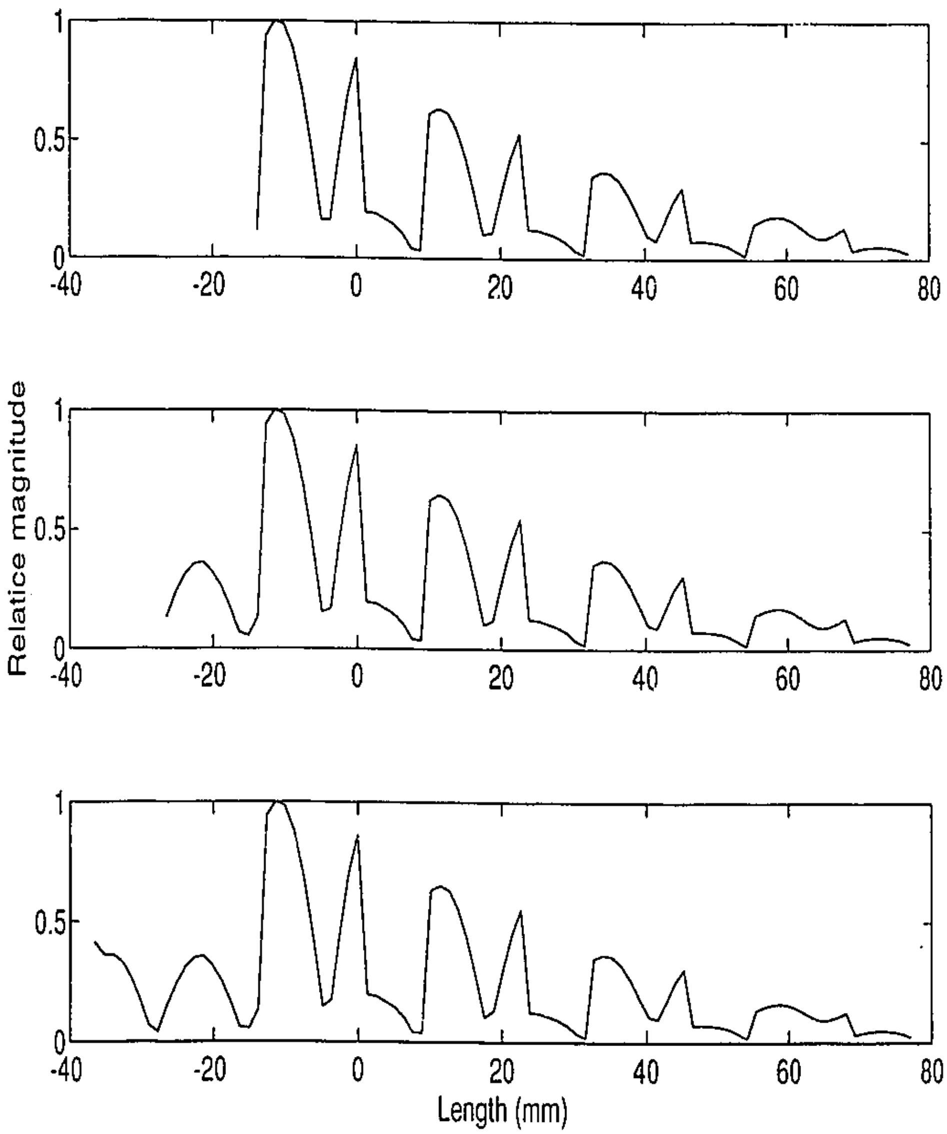


Figure 4.28: The current distribution on the center lines of the series-fed microstrip arrays with $L = 17mm, 25mm, 38mm$, respectively.

The standing current waves exist on the feed networks because of the mismatch between the feed lines and the patch elements. Ideally, the impedance of the elements should be matched to the feed line to prevent standing waves. In practice, there are always some uncertainties with regard to the matching conditions due to imperfections in the simulations, design and fabrication, which, in turn, lead to more or less standing waves. Based on these current distributions, the effect of the feed network on the radiation pattern of the whole array can be predicted by calculating the radiation field from different parts of the structure. The formulations are given below [60].

$$E_\theta = -j60k\frac{e^{-jkR}}{R}[\cos\phi\tilde{I}_x(\theta,\phi)+\sin\phi\tilde{I}_y(\theta,\phi)]P_\theta(\theta,\phi) \tag{4.86}$$

$$E_\phi = +j60k\frac{e^{-jkR}}{R}[\sin\phi\tilde{I}_x(\theta,\phi)-\cos\phi\tilde{I}_y(\theta,\phi)]P_\phi(\theta,\phi), \tag{4.87}$$

where

$$P_\theta = \frac{\cos\theta\sqrt{\epsilon_r-\sin^2\theta}\sin(k_0\sqrt{\epsilon_r-\sin^2\theta}z_0)}{\sqrt{\epsilon_r-\sin^2\theta}\sin(k_0\sqrt{\epsilon_r-\sin^2\theta}z_0-j\epsilon_r\cos\theta\cos(k_0\sqrt{\epsilon_r-\sin^2\theta}z_o)} \tag{4.88}$$

$$P_\phi = \frac{\cos\theta\sin(k_0\sqrt{\epsilon_r-\sin^2\theta}z_0}{\cos\theta\sin(k_0\sqrt{\epsilon_r-\sin^2\theta}z_o)-j\sqrt{\epsilon_r-\sin^2\theta}\cos(k_o\sqrt{\epsilon_r-\sin^2\theta}z_0)}. \tag{4.89}$$

Here, $\tilde{I}_x, \tilde{I}_y$ are the Fourier transforms of current distribution defined as

$$\tilde{I}_\zeta(\theta,\phi)=\int\int J_\zeta(x',y')e^{j\vec{k}_s\cdot\vec{r}'}dx'dy' \qquad \zeta: x \text{ or } y \tag{4.90}$$

with

$$\vec{r}'=x'\hat{x}+y'\hat{y}+z_0\hat{z}; \qquad \vec{k}_s=k(\sin\theta\cos\phi\hat{x}+\sin\theta\sin\phi\hat{y}+\cos\theta\hat{z}) \tag{4.91}$$

and z_0 denotes the thickness of the dielectric substrate.

Fig.4.29 shows the E-plane radiation pattern for a series-fed array, compared with the experimental results. Good agreement between the experimental data and the theoretical data is achieved. As is pointed out in [53], the effect of the feed

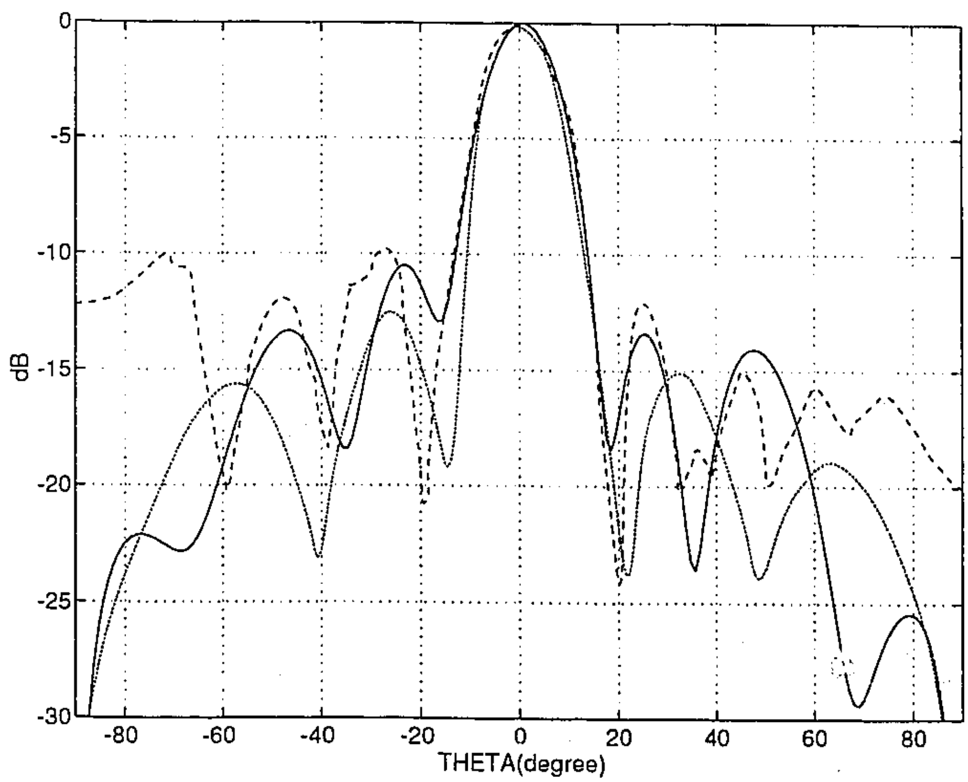


Figure 4.29: The E-plane radiation patterns of the series-fed microstrip array. Solid line: simulation result, dashed line: experiment result, dot line: the simulated result with no feed network.

networks on the radiation patterns varies with the lengths of the feed lines. This is a result of the change of the standing current waves on the feed lines. For comparison, the radiation patterns of arrays with different lengths of feed lines are shown in Fig.4.30. Fig.4.31 shows the computed radiation patterns of the feed network alone, i.e., with the radiation from the patches suppressed. These patterns

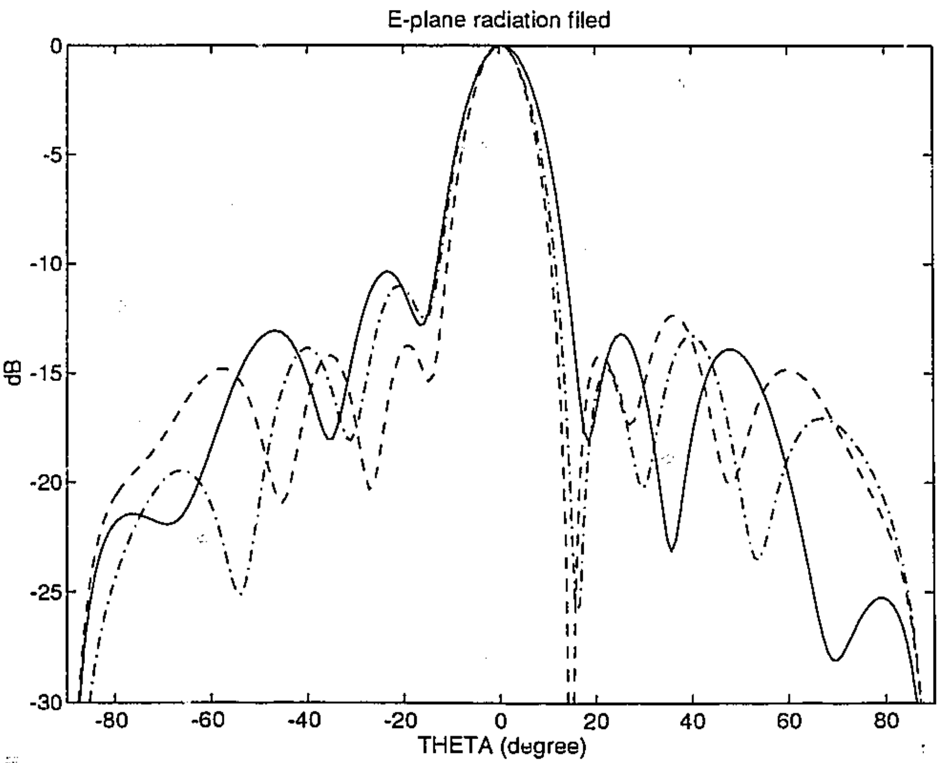


Figure 4.30: The E-plane radiation patterns of the series-fed microstrip arrays with different lengths of the feed lines. Dashed-dot line: $L = 17mm$, dashed line: $L = 25mm$, solid line: $L = 38mm$.

are compared with the envelope of the side-lobe pattern of the patches alone. It

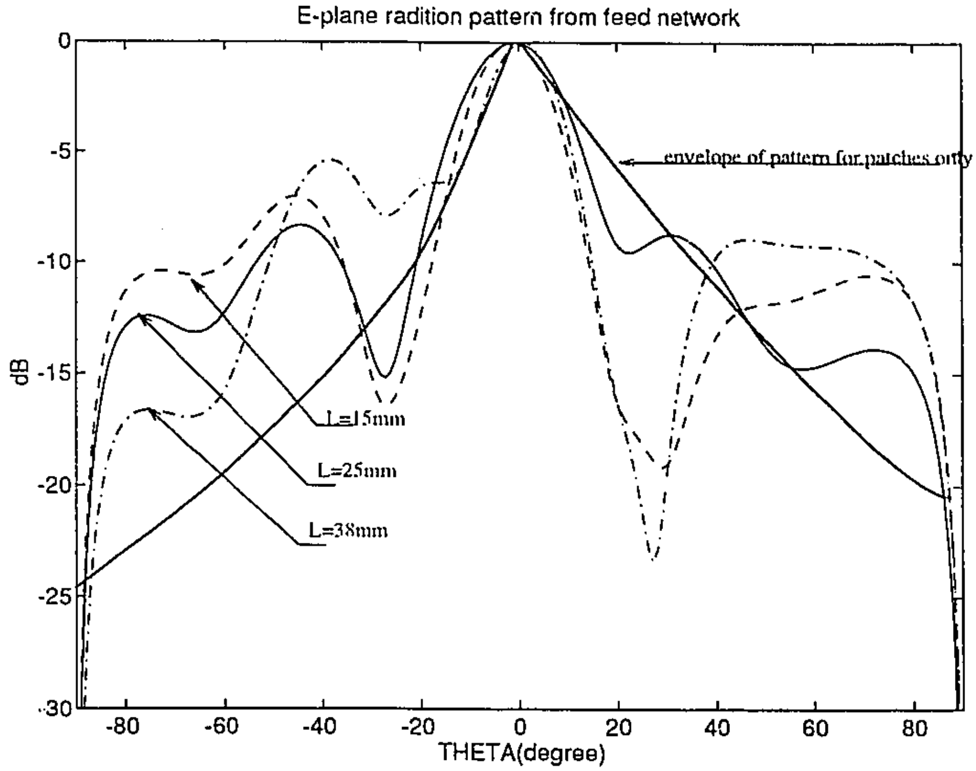


Figure 4.31: Radiation patterns for different lengths of the feed networks.

can be seen that the feed side-lobes change with the length of the feed line, and in some cases, they are of the order of 10dB higher than that of the patch radiator. Therefore, the increased side-lobe level (SLL) of the whole array, i.e. that part of the radiation which is greater than that from an array without a feed network, is attributed to spurious radiation from the feed network.

Next, we simulate a new array. It is a 2×2 corporate-fed patch array as shown in Fig.4.32. The source is located at the center of the feed network in which a coaxial probe is fed from the back of the substrate. Figs.4.33 and 4.34 show the current distribution on the patch and feed lines directly connected with the patches. The measured and calculated H-plane radiation patterns of this structure

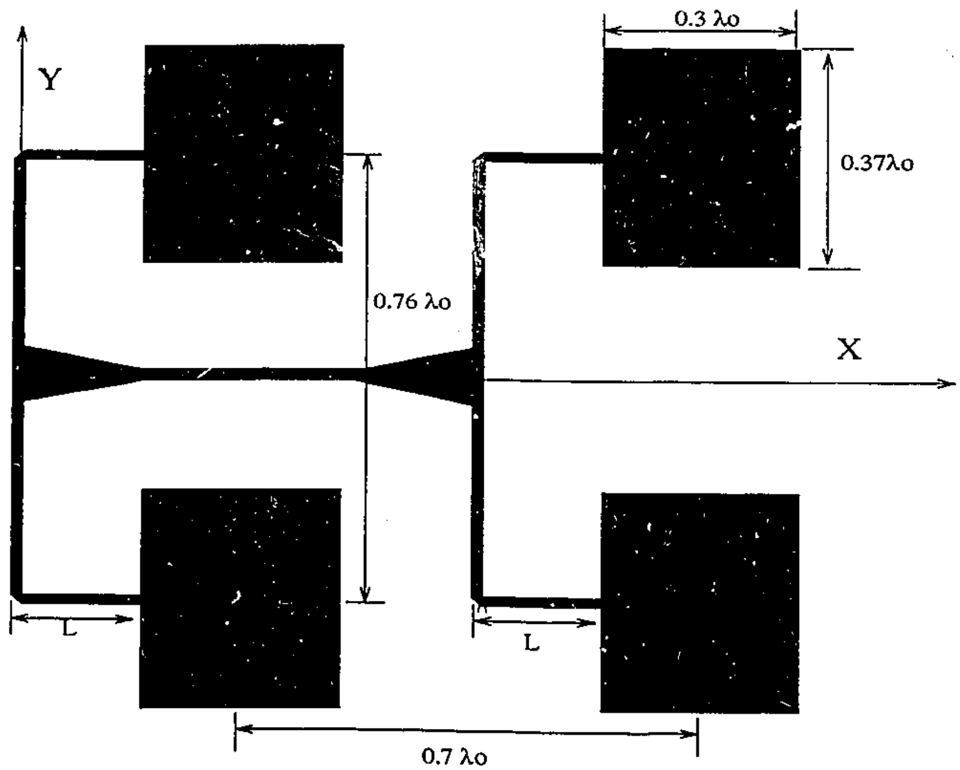


Figure 4.32: The geometry of a 2 × 2 corporate-fed microstrip array.

are shown in Fig.4.35. They are also in good agreement. To see the effect of the feed networks, the calculated E-patterns of the arrays with different lengths of feed lines are shown in Fig.4.36. It is found that the array with $L = 0.1\lambda_0$ has the smallest SLL. This suggests that the feed lines should be kept as short as possible to reduce the level of the spurious radiation. On Fig.4.35, the cross "+" represents the patterns with the lengths of the feed lines changed. Since the currents on feed lines are mostly flowing in the x-direction, the change of the lengths of the feed lines has a minor effect on the H-plane patterns (only a tiny difference on the

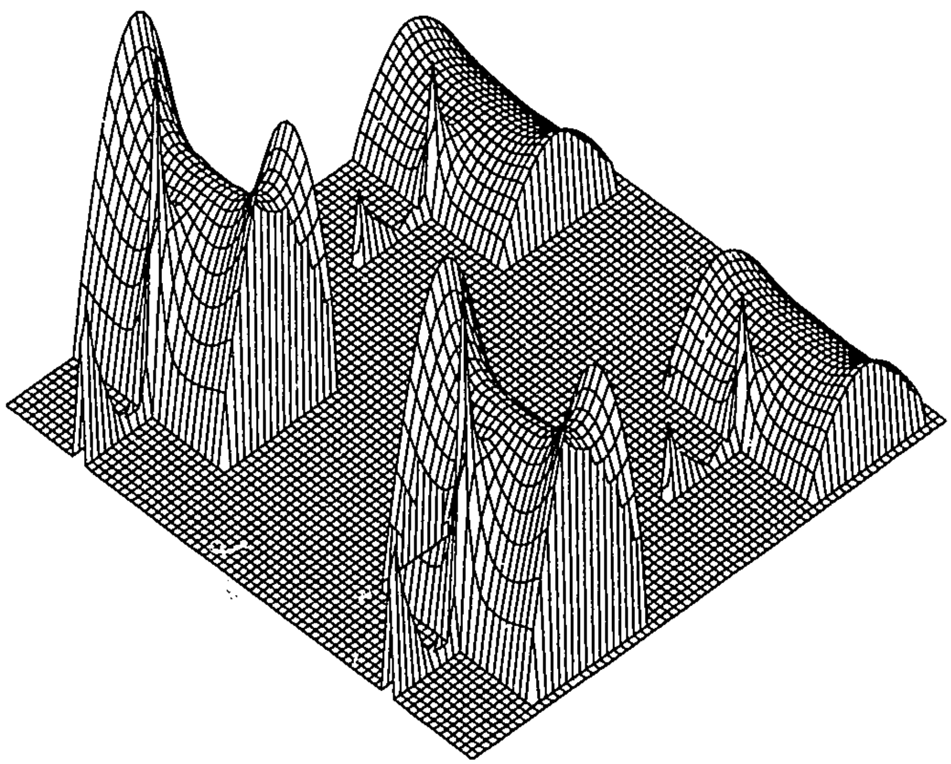


Figure 4.33: The x-directed current distribution of the 2×2 microstrip corporate feed array. Note that the current on the central part of the feed line is not draw here.

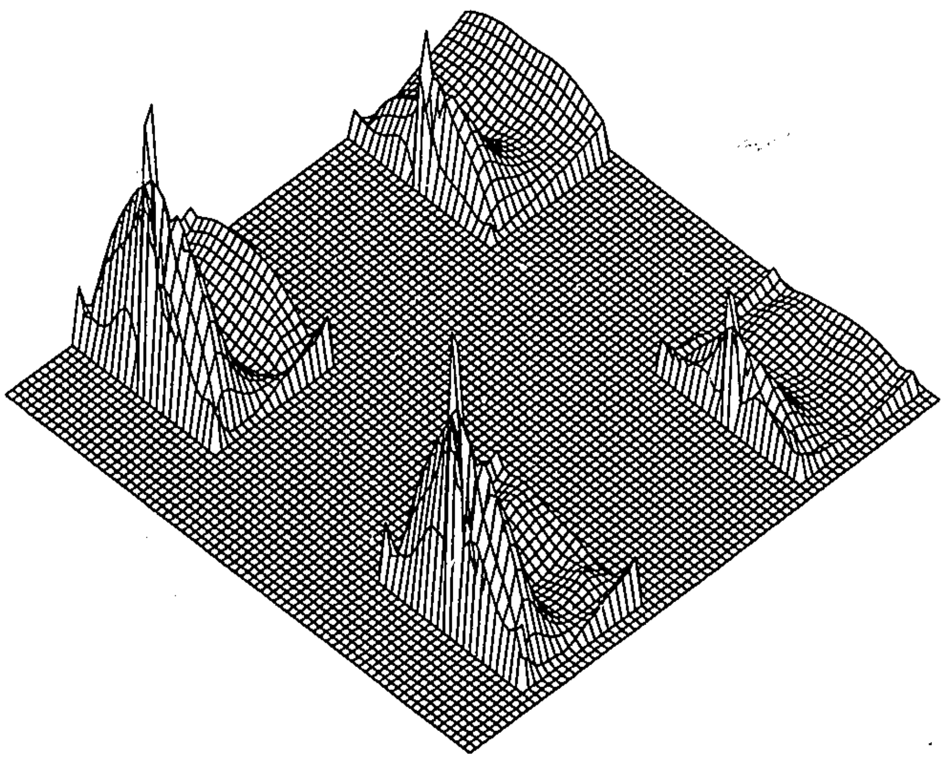


Figure 4.34: The y-directed current distribution of the 2 × 2 microstrip corporate feed array. Note that the current on the central part of the feed line is not draw here.

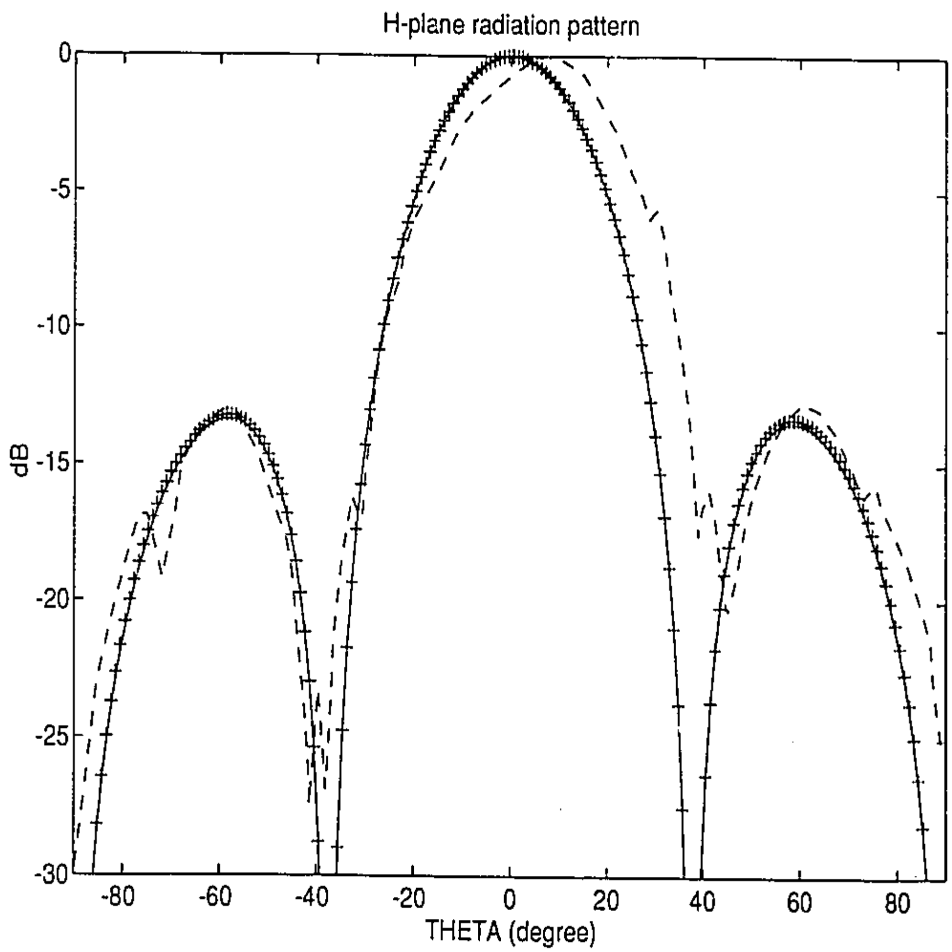


Figure 4.35: The H-plane radiation pattern of the 2×2 corporate feed microstrip planar array. Solid line : the simulation result; dashed line: measured result; cross +: the simulation result with the length of the feed line changed.

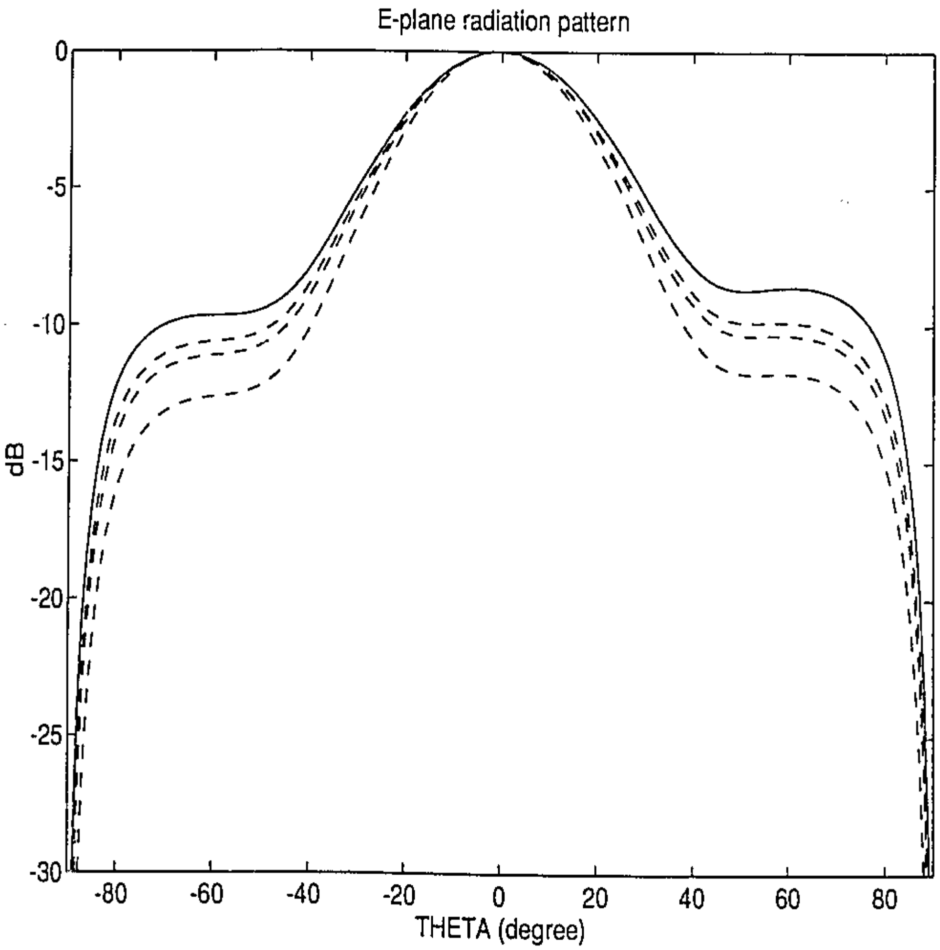


Figure 4.36: The E-plane radiation pattern calculated from the current distribution with different lengths of the directly connected feed lines. The patterns from with highest SLL to lowest SLL are corresponding to the lengths $L = 0.4\lambda_0, 0.3\lambda_0, 0.2\lambda_0, 0.1\lambda_0$, respectively.

side-lobe).

In the following, larger microstrip planar arrays are analyzed. One is an 8×4 corporate feed array composed of 8 branches of the 4-element series-fed arrays, shown in Fig.4.37. A 3-dimensional view of the x- and y-directed current

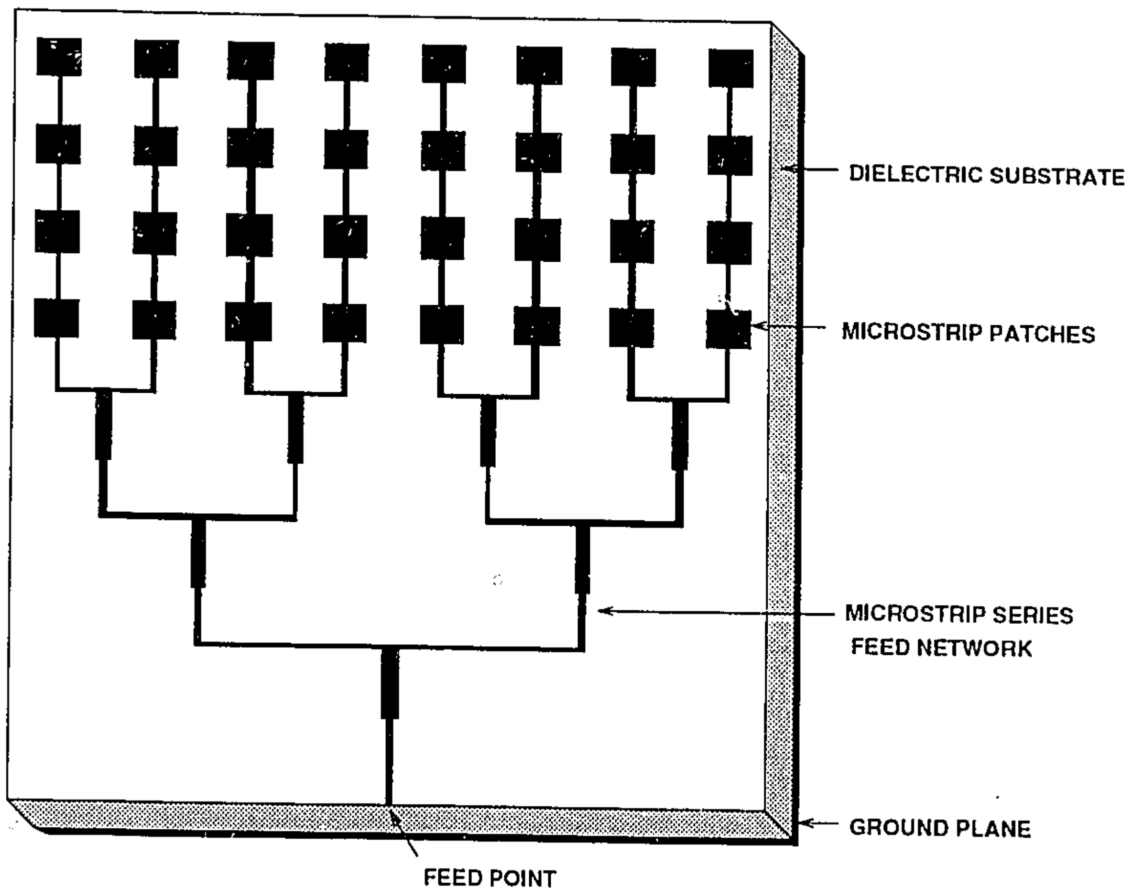
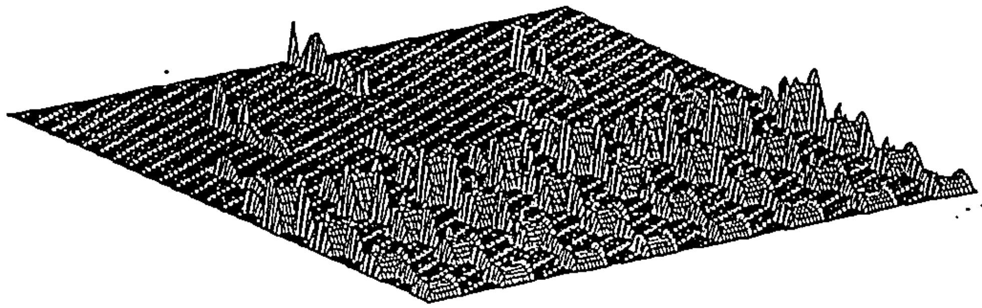


Figure 4.37: A geometry of a microstrip planar array with a corporate feed network.

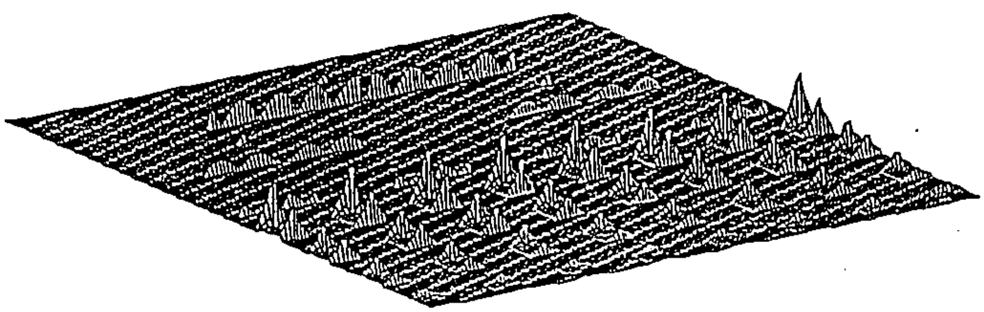
distributions are depicted in Fig.4.38. The standing current waves, caused by the mismatched condition, are clearly shown on every part of the feed networks.

Figures.4.39 and 4.40 give the radiation patterns in E- and H-planes compared with the patterns calculated for arrays without feed networks. It is seen that feed network affects the H-plane pattern only in the region of its side-lobes and the effects are symmetric, while its effects on the E-plane patterns are more destructive, where even the main-lobe is degraded, and the effects are asymmetric.

As another example, a 4×4 corporate planar array, composed of four of the 2×2 subarrays shown in Fig.4.32, is simulated. The configuration is illustrated in Fig.4.41. The source is at the center of the feed network, driven by a coaxial line from the back. The effect of the whole feed network on the array's radiation pattern can be predicted. For simplicity, the effect due to the short feed lines which are directly connected with the patch elements are analyzed. Figs.4.42 and 4.43 show the x-and y-directed current distributions on the patch elements and feed lines (the current on the other parts of the feed network are suppressed).



(A)



(B)

Figure 4.38: The current distributions on the 8 × 4 corporate feed array. (A):

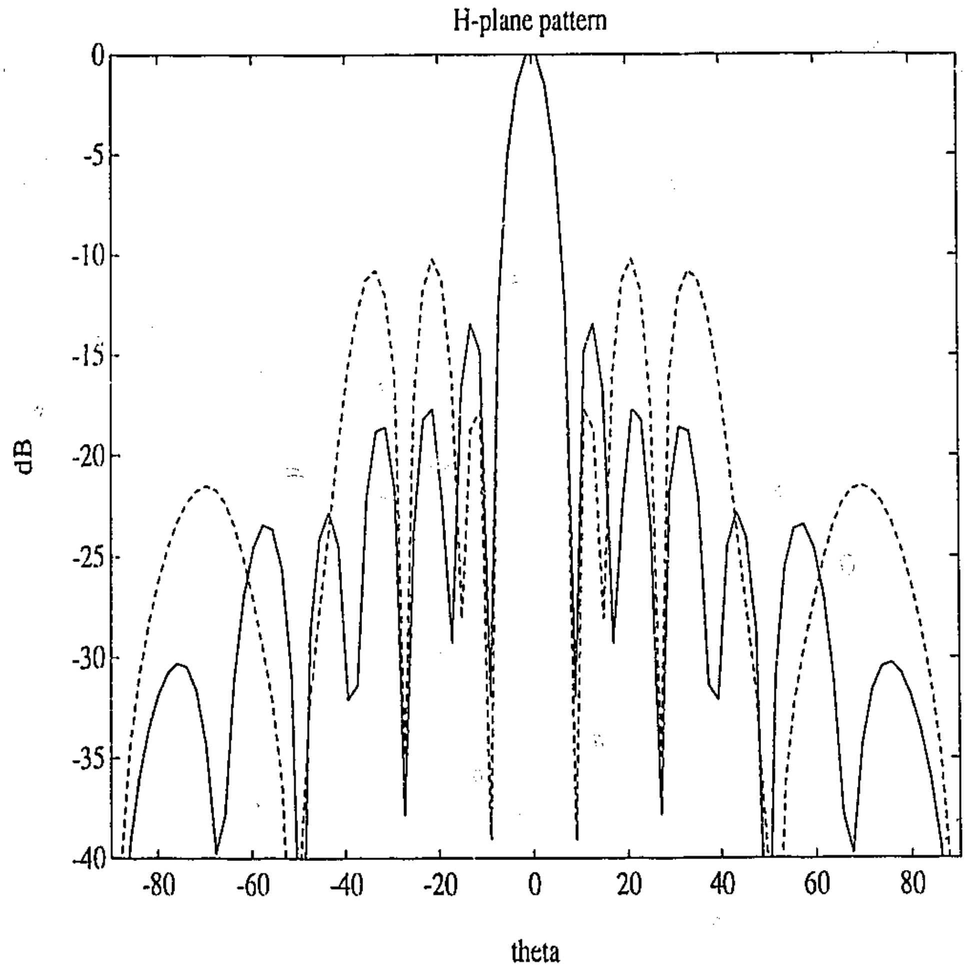


Figure 4.39: The H-plane radiation pattern of the 8×4 corporate feed array. Solid line : pattern without feed network; dashed line: pattern with feed network.

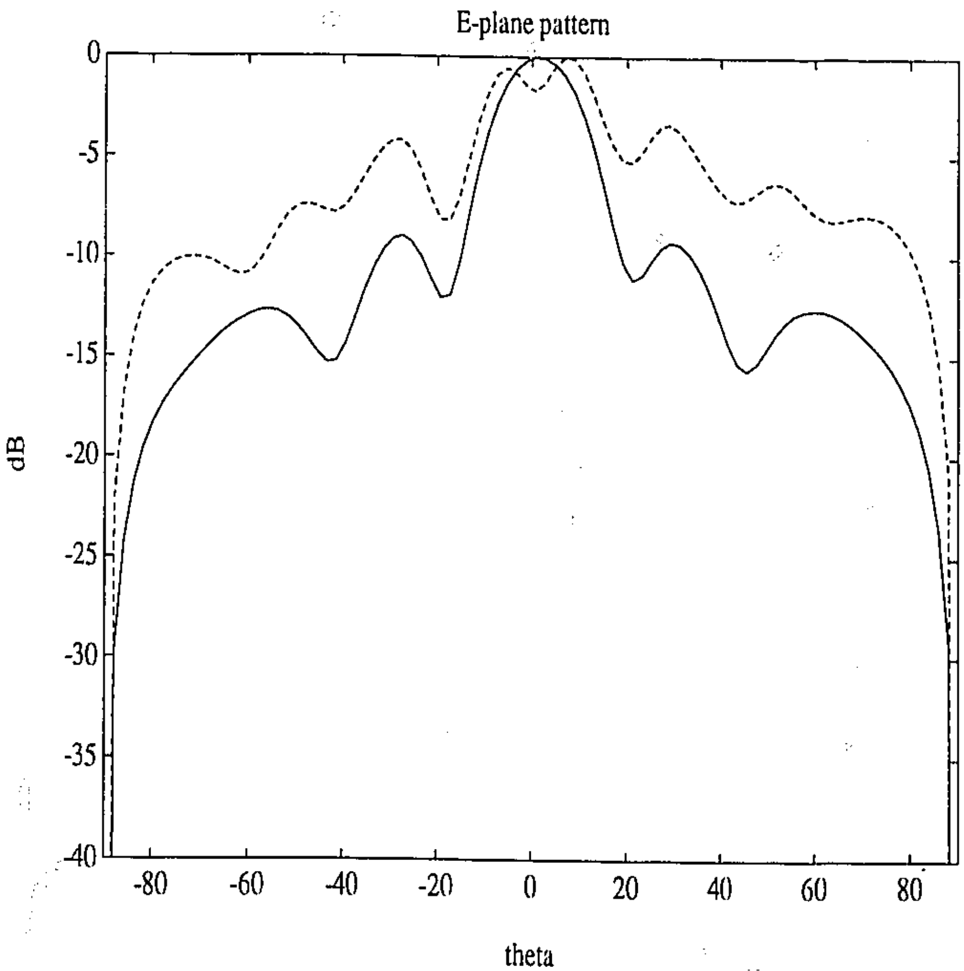


Figure 4.40: The E-plane radiation pattern of the 8×4 corporate feed array. Solid line : pattern without feed network; dashed line: pattern with feed network.

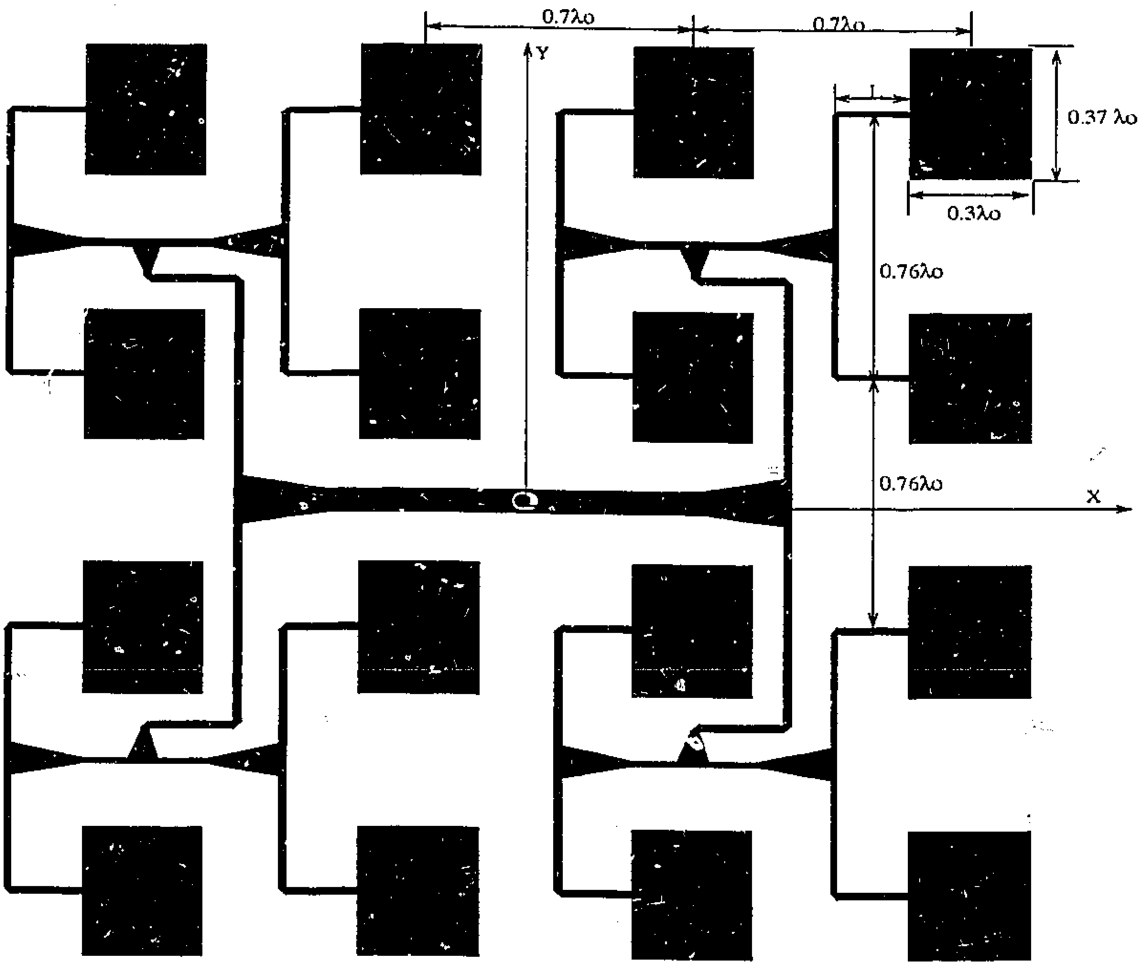


Figure 4.41: The geometry of a 4 × 4 corporate feed microstrip feed array.

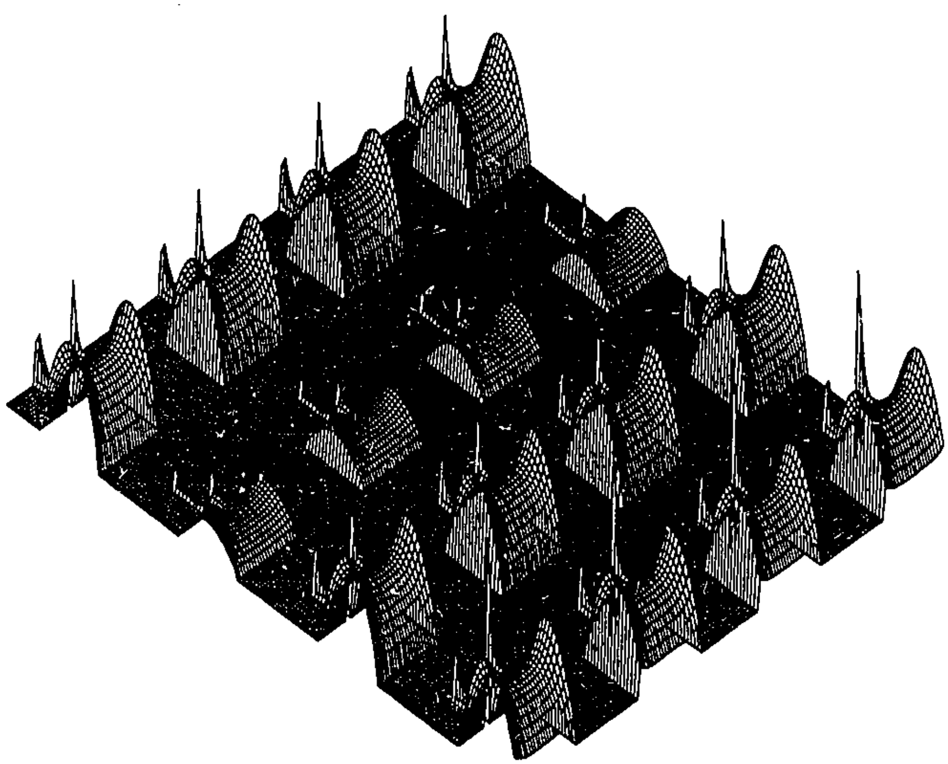


Figure 4.42: The x-directed current distribution on the corporate feed array shown in the previous figure.

The radiation patterns for this structure are calculated using the current distributions shown above. Figs.4.44 and 4.45 show the E- and H-plane radiation patterns of the arrays with different lengths used for the feed lines ($L = 0.4\lambda_0, 0.3\lambda_0, 0.2\lambda_0, 0.1\lambda_0$). Since the feed lines support only x-directed components of current, only the E-plane patterns for the antenna are affected. It is also seen that the shorter the feed line, the lower the level of the spurious radiation.

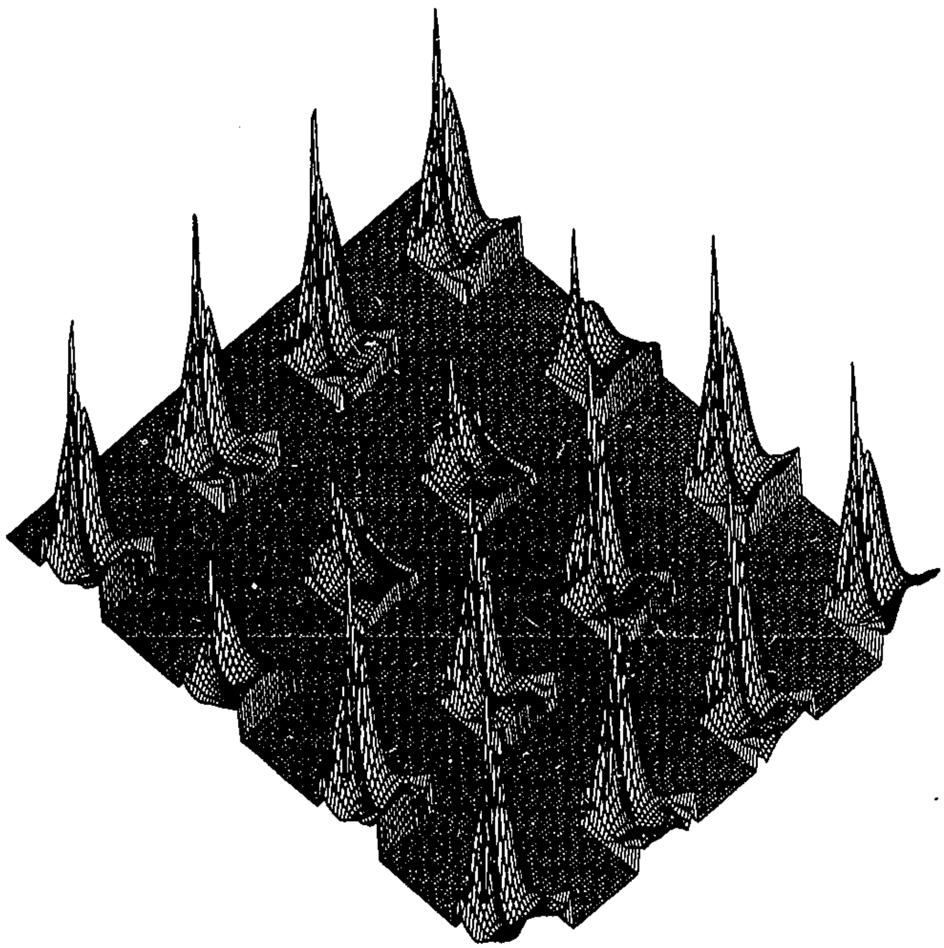


Figure 4.43: The y-directed current distribution on the 4×4 corporate feed array.

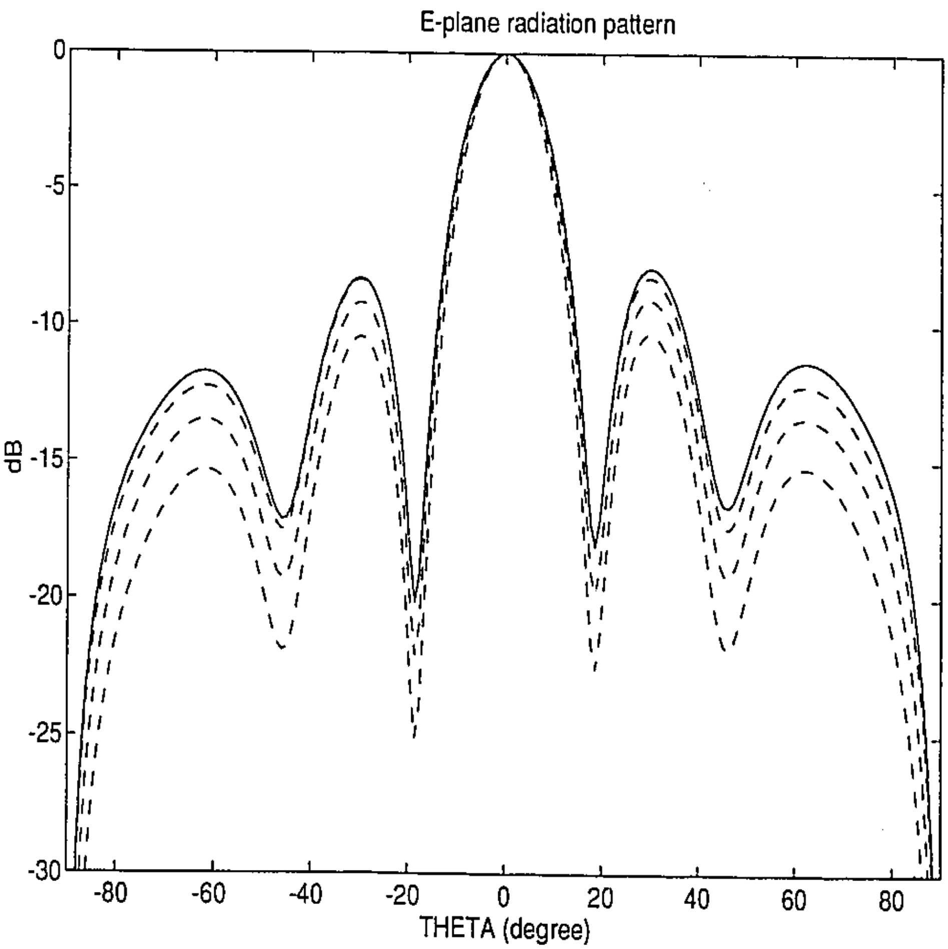


Figure 4.44: The E-plane radiation pattern calculated from the current distribution with different lengths of directly connected feed lines (from highest SLL to lowest SLL, $L = 0.4\lambda_0, 0.3\lambda_0, 0.2\lambda_0, 0.1\lambda_0$), respectively.

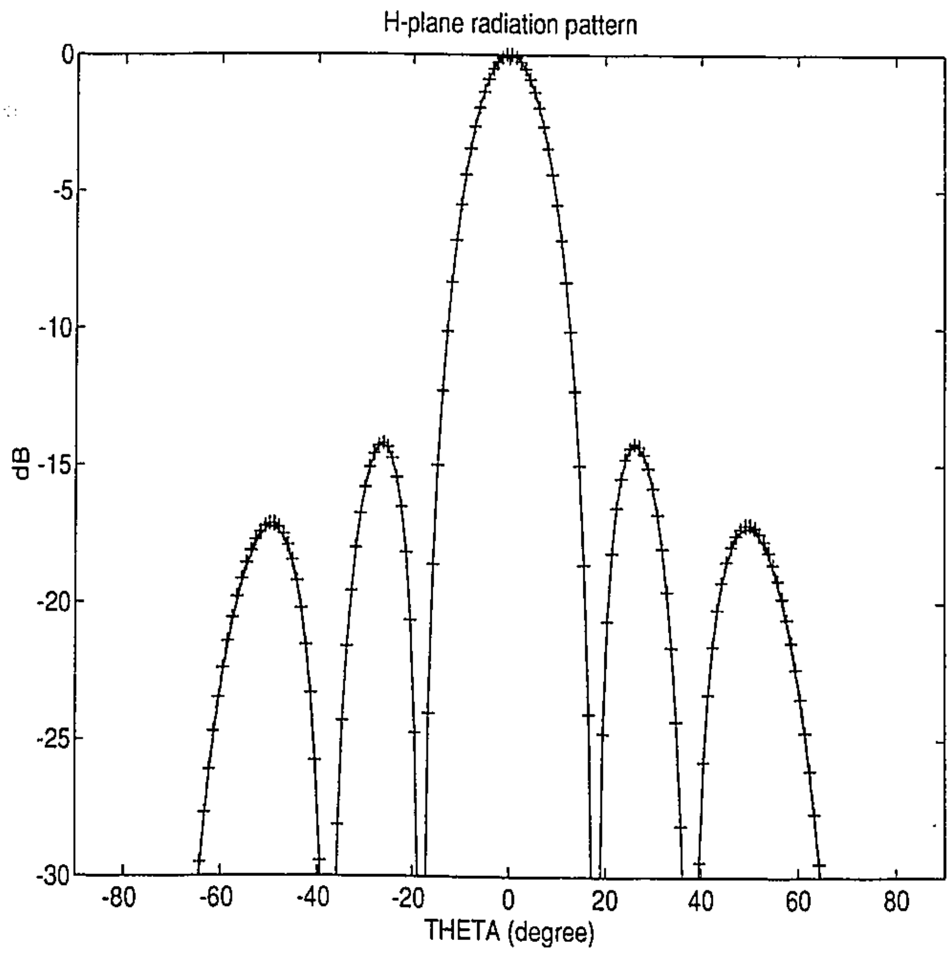


Figure 4.45: The H-plane radiation pattern calculated from the current distribution with different lengths of directly connected feed lines.

4.4 Modeling Study of Microstrip Reflectarrays

The most important advantage of the CG-FFT method is the potential that it has for solving electrically large EM scattering and radiation problems. Some applications of CG-FFT for large scattering problems have been reported, but very few applications of this method to the analysis of large microstrip arrays have been reported in the literature so far. To demonstrate the strength of the hybrid CG-FFT, a modeling study of large microstrip reflectarrays is presented in this section.

4.4.1 The Concept of Microstrip Reflectarray

Mobile and satellite communication systems require high gain antennas which are compact in size and light weight. Antenna technologies that can be used to develop physically small antennas are of considerable interest to antenna engineers. The microstrip reflectarray (MRA in short) has been suggested as a possible antenna for mobile and satellite communications (see Fig. 4.46) [61] [68].

The use of reflector antennas for beam forming is analogous to reflectors used in optics - the reflector surface is designed and fabricated so as to compensate for the phase differences of the fields traveling from the feed to different points on the reflector. An outgoing coherent phase-front is thereby generated by the reflector. To reach the same goal, the microstrip reflectarray (Fig. 4.47) uses microstrip patches connected to open or short transmission lines to form a directional beam. The patches serve as re-radiators, and the connected microstrip transmission lines (we call them tails) act as phase shifters. The tails are adjusted so as to generate a coherent field in the wanted direction. For example, for a parabolic reflector

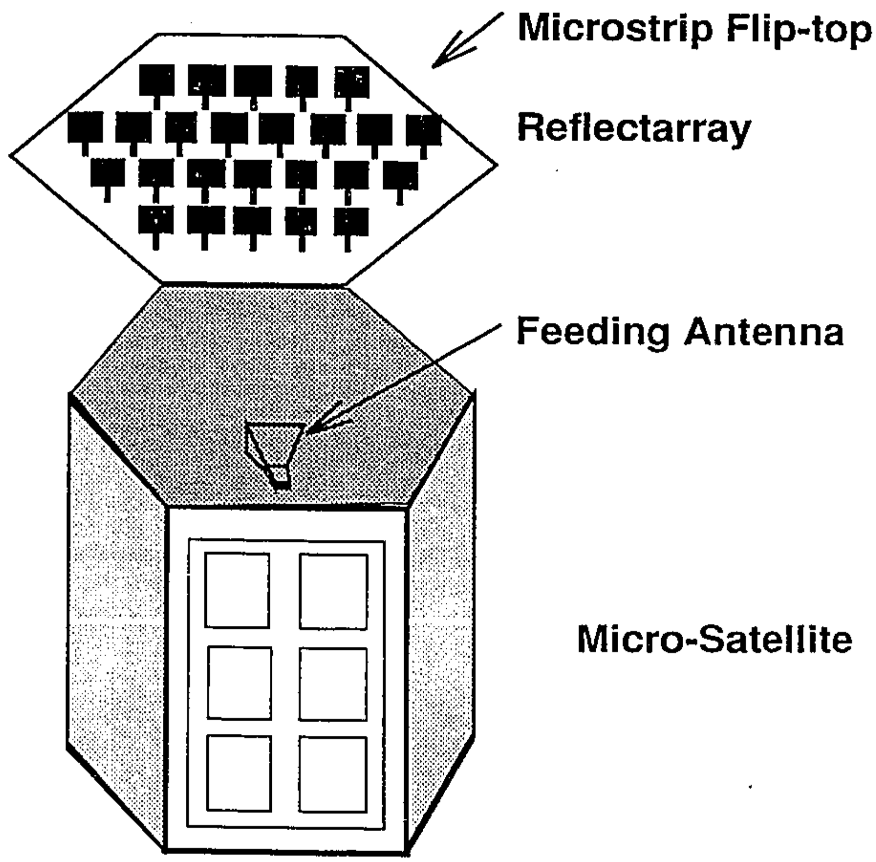


Figure 4.46: The use of microstrip reflectarrays for micro-satellite for satellite communications.

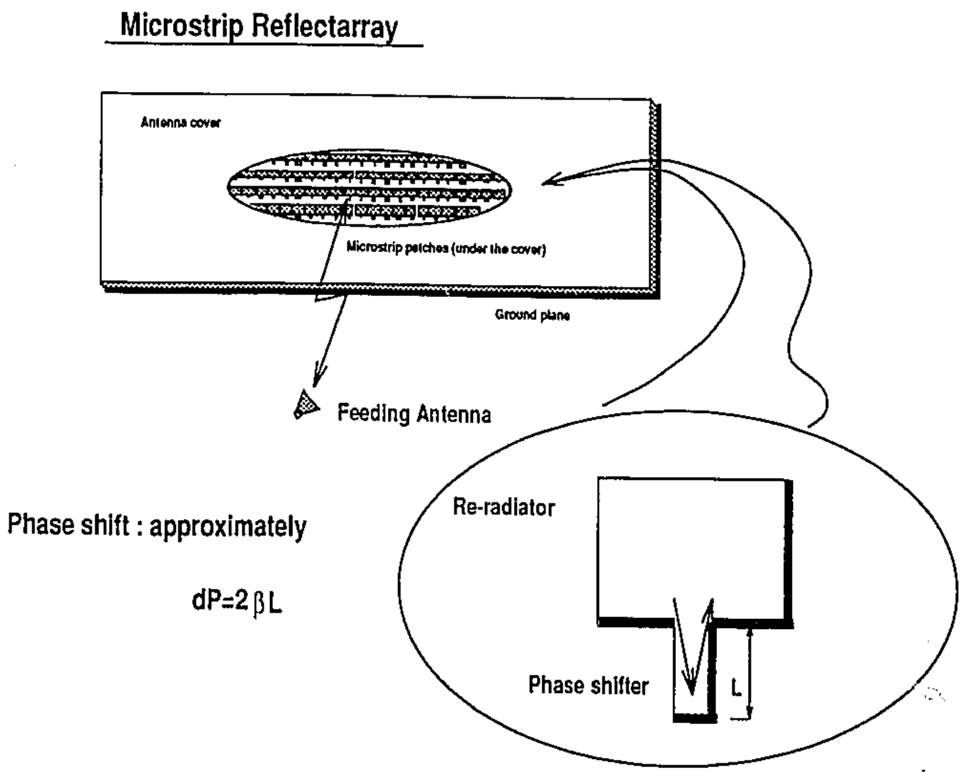


Figure 4.47: The composition of a microstrip reflectarray and the patch elements which operate as both a phase shifter and re-radiator.

antenna (Fig.4.48),

$$FM + MM' = FO + OO', \quad F \text{ is the focus of the parabolic reflector}, \tag{4.92}$$

while for the microstrip reflectarray

$$FP + PP' \neq FO'' + O''O'. \tag{4.93}$$

However, with a good design, we can have

$$FP + PP' = FO'' + O''O' + dp \tag{4.94}$$

where dP denotes the equivalent electrical length that corresponds to the phase shift imported by the patch.

Compared with reflector antennas, the microstrip reflectarray has the following merits:

- Owing to its flat structure, MRA can achieve larger sizes with less structural distortion, thereby easing the problem of mounting the antenna, as well as leading to a more cost effective antenna.

- It is possible for MRA to have a larger offset main beam and feed point because the offsets are achieved by tuning the electrical lengths of the tails rather than the shape of the reflectors.

- Electronic beam scanning is possible if phase shifters are incorporated with the tails.

Compared with the microstrip planar arrays, the MRA has the advantage of not requiring a complex corporate feed network. This means that the MRA

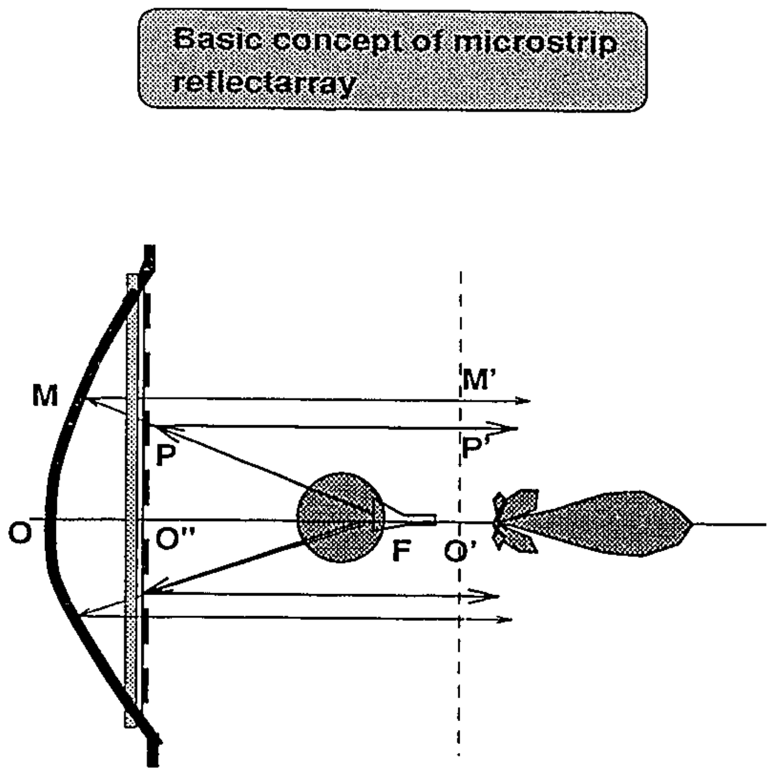


Figure 4.48: Illustration of the concept of the microstrip reflectarray - comparison with a reflector antenna.

may have less system complexity, less insertion loss, higher efficiency and a larger power handling capability.

Following the introduction, by Munson et al, of the concept of microstrip reflectarrays in 1987 [62], only a few feasibility studies have been carried out. They have been primarily experimental in nature [63]. This is due to the fact that, as of yet, virtually no design tools have been developed for these types of structures. To add to this challenge, it should be noted that the structures are usually fairly large. In [64], an analytical approach, combining basic antenna array theory and the GTD, has been proposed. This approach can give us some basic understanding and a rough prediction of the performance of this type of antenna. Another approach based on the Method of Moments and Floquet theory for infinitely large arrays has also been proposed in [65] [66]. This approach can yield much more accurate results.

In this section, we present a modeling study of an MRA. This work is carried out using the hybrid CG-FFT method. The efficiency of this technique allows us to easily deal with very large MRAs. Based on this full-wave analysis technique, we have studied the re-radiated field phase shift vs. length of the microstrip tails, magnitude of current distributions vs. the length of the tails, the current distribution on the entire array and the radiation pattern of the MRA. To validate our model, several trial designs have been done. The experimental results verify our theoretical studies.

4.4.2 Analysis

To accurately analyze a very large microstrip reflectarray, in particular, " to accurately predict the backscattered fields from thousands of non-uniformly illuminated microstrip patches with un-equal lengths of microstrip transmission lines, a very complex analysis technique needs to be developed [64]. " The Conjugate-Gradient Fast Fourier Transform Method (CG-FFT) is very suitable for analyzing large microwave structures. When combined with the discrete image technique [24], the CG-FFT becomes a very robust technique for analyzing very large microstrip antenna arrays [23]. Based on this technique, an MRA design procedure is proposed [67], which is given in flow chart form in Fig.4.49. When implementing this procedure, the hybrid CG-FFT is utilized to simulate a large microstrip reflectarray, as well as to predict the phase shifts of single patches with different lengths of tails, all in an array environment [69]. In the simulation, the microstrip structure is discretized into rectangular small cells. The roof-top functions are used to expand the current on the structure. The incident field $\vec{E}^{inc}$ is defined by the characteristics of the feed antenna. When the type and the location of the feed are identified, it is not difficult to calculate the incident fields precisely (magnitudes, phases and glazing angles) at every position within the reflectarray. Then the hybrid CG-FFT described in Chapters 2 and 3 is applied to calculate the current distribution. Following the calculation, the current $\vec{J}$ on a radiating element (patch plus tail) is Fourier transformed to provide the phase of the field that is re-radiated from the element. Alternately, the radiation pattern is calculated by combining the reflected field from the ground plane and the field re-radiated from the patches using the formulation given in equation 4.86 - 4.91.

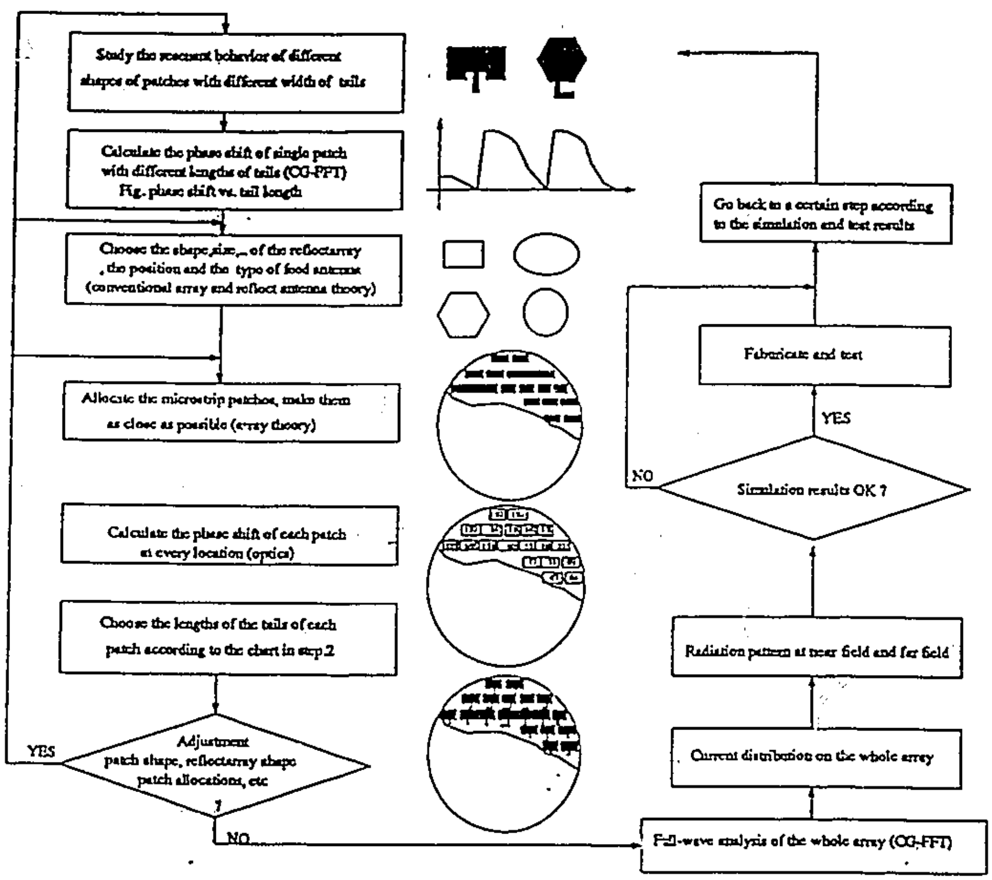


Figure 4.49: A proposed procedure for microstrip reflectarray design.

4.4.3 Results

We have simulated and tested some trial X-band linear microstrip reflectarrays, with different numbers of patches. Each patch was resonant at 12 GHz. The ϵ_r of the substrate used was 2.2, and its thickness was 1.59mm.

First, a critical characteristic of the MRA, i.e., the relationship between the phase shift of the reflected field from a single patch and the length of the connected tail is investigated. This is presented by the solid line in Fig.4.50. To show the effect of mutual coupling, a single patch in an array environment has been analyzed, and the phase shift relation is shown in Fig.4.50 (dashed line). To study the effect of different shaped patches, the phase shift of patches with indented tails, which can be used to carry out an impedance match between the tail and the patch, are also simulated(Fig.4.50, dotted line). The indentations are 1.2mm vertically and 0.9mm horizontally.

In the simulations, we found that the current distributions on the patches are affected by the attached tails. This behavior is due to the fact that the impedance looking into the tail, at the junction between the patch and its tail, varies with the length of the tail. For instance, the impedance should be near zero when the tail length is $\frac{\lambda_g}{4}$. The current distribution on the patch for this case should be close to zero because of the short-circuiting effect of the tail. As the length either increases or decreases with respect to $\frac{\lambda_g}{4}$, the current increases monotonically. This is known as the $\frac{\lambda_g}{4}$ effect. These relationships are shown in Fig.4.51.

A simple 23×1 linear microstrip reflectarray is simulated. The co-polar current distribution is shown in Fig.4.52. The radiation pattern is calculated from

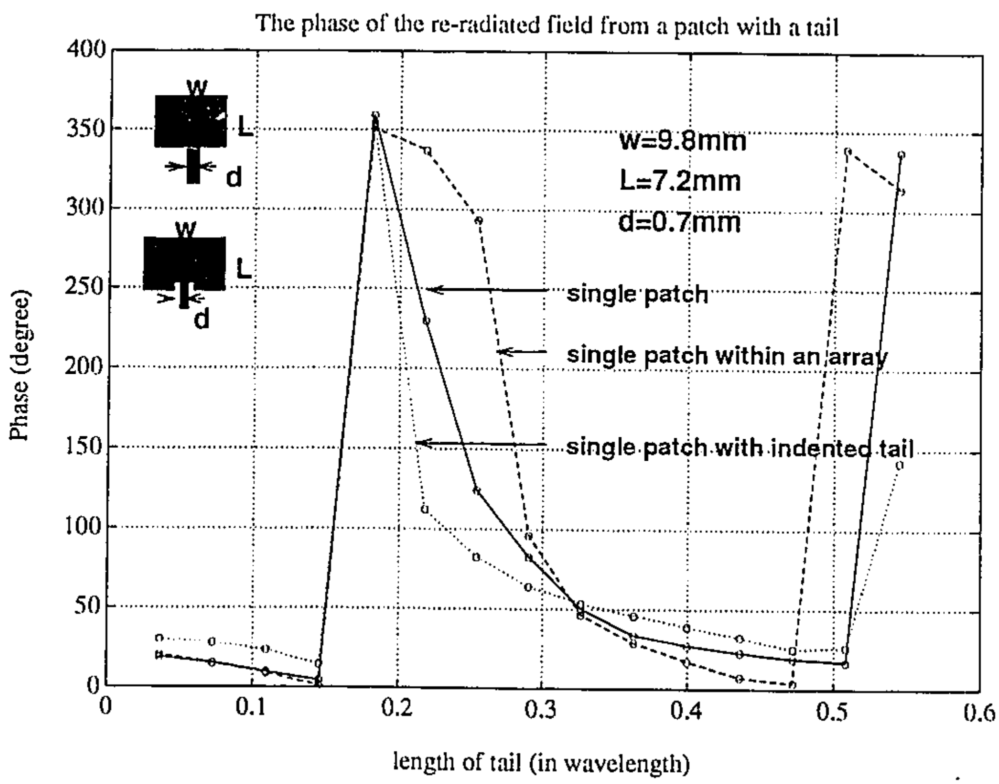


Figure 4.50: The phase shift of the re-radiated field vs. the length of the tail from a single microstrip patch element.

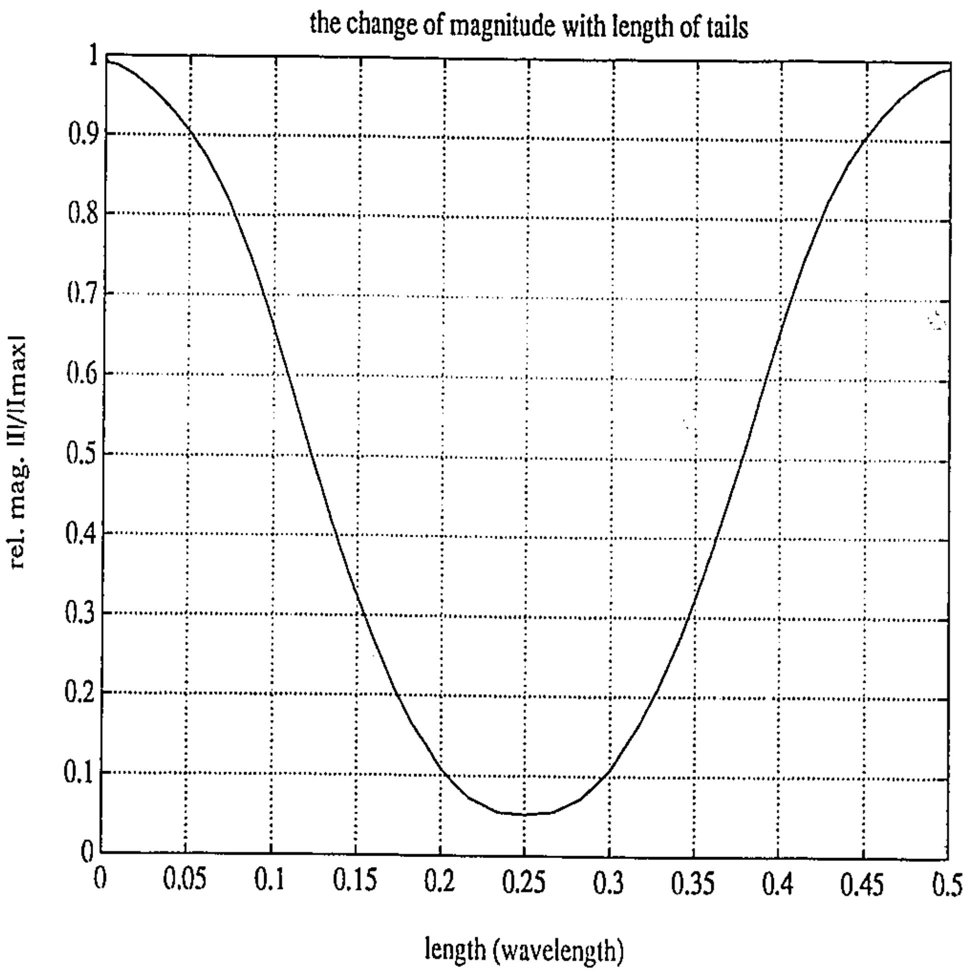


Figure 4.51: The change of the average magnitude of the normalized current vs. the length of the tail on a single microstrip patch element.

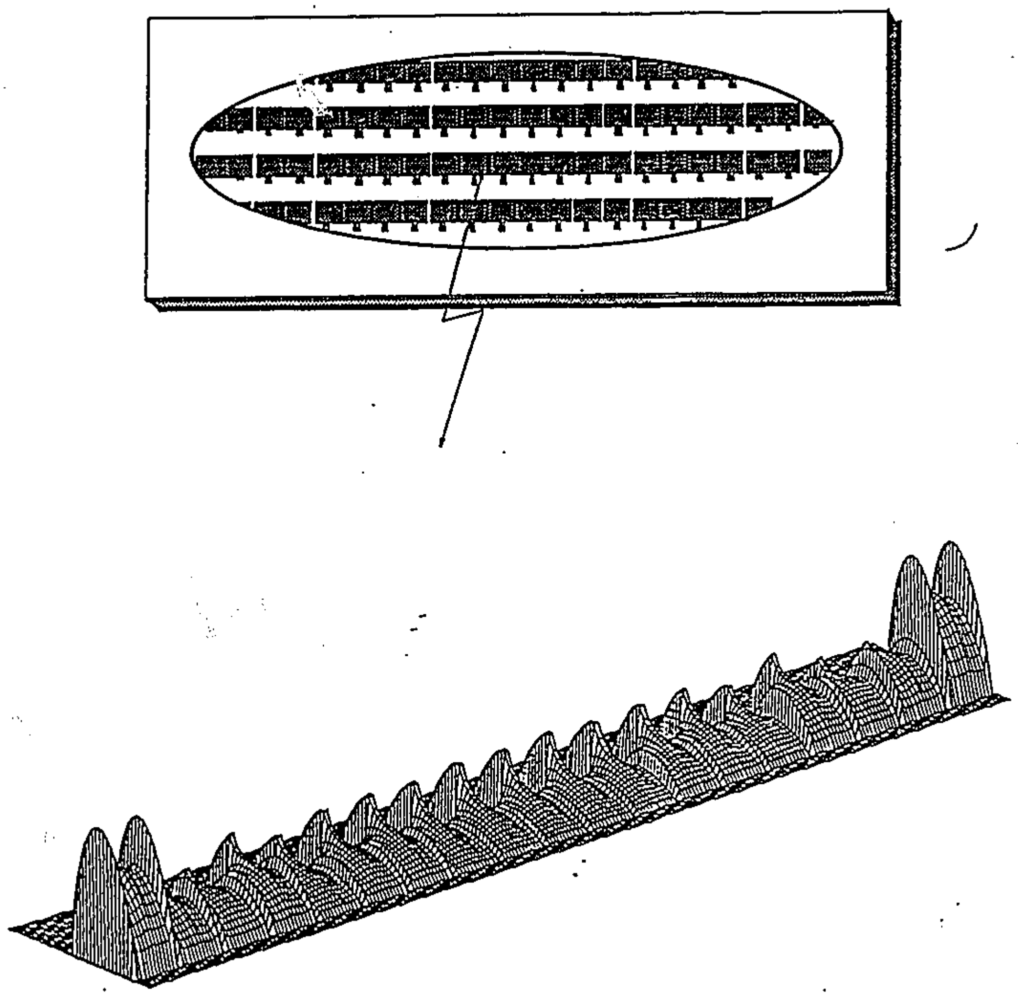


Figure 4.52: The co-polar current distribution on a linear microstrip reflectarray. Note that the magnitudes of the current on the central part are smaller than those near either edge of the array. This is attributable to the high SLL of the radiation pattern of this array.

the current distribution. The H-plane (focused plane) pattern is shown in Fig.4.53. The main beam is seen to be focused in the designed direction. The measurements on the array were carried out in an anechoic chamber. The measured pattern (dotted line in Fig.4.53) is compared with the simulated pattern and reasonably good agreement is observed in this figure. Due to the fact that we did not have access to a proper 12GHz low-noise-amplifier, the pattern measured outside of 50^0 is dominated by noise.

It is seen that the currents on the central patches are smaller than those on the patches located near the sides of the array. This V-shape current distribution over the array results in relatively high side-lobe levels. This is because of: (1) the edge effect of the backscattering field from the arrays; (2) the relationship between the length of the tails and the current magnitude along the array.

To overcome the destructive edge effect and tail effect, we propose the following scheme for realizing improvements:

- Add some dummy elements at both sides of the array. In our studies, we found that if we put some patches with $\frac{\lambda_g}{4}$ tails at both sides of the array, the short-circuiting effect and the edge effect can cancel each other.

- Combine and use two techniques for tuning the array. In [65], it was found that the phase of the re-radiated field can be controlled by changing the length of the patch. So we propose combining the two techniques when tuning the patches, i.e., changing the lengths of the tails, as well as the sizes of the patches. In this way, we can achieve an optimal current distribution and the phase shifts required to get both low side-lobes and a well defined main beam pattern.

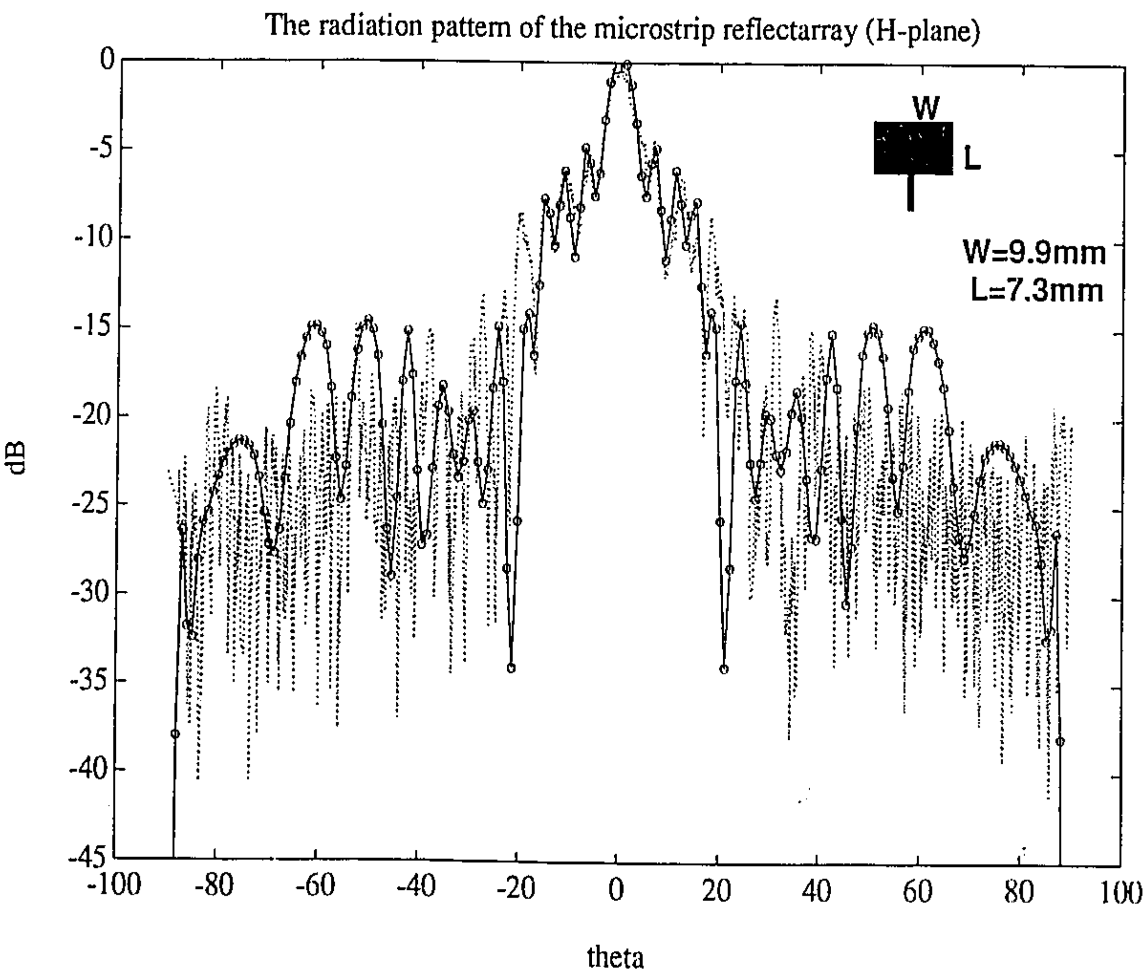


Figure 4.53: The H-plane radiation pattern of the linear microstrip reflectarray. Solid line: simulated result, dotted line: experimental result.

In Figs.4.54 and 4.55 are given the results of the simulations carried out on the adjusted microstrip array, where we start with the previously designed MRA and apply the above described adjustments. The early results are given in Figs.4.52 and 4.53. When a comparison is made with the results in Fig. 4.53, we see that some improvements have been made in the current distribution and side-lobe levels of the modified MRA. It is clear that further refinements are required to improve the techniques currently available for designing and testing large two-dimensional MRAs.

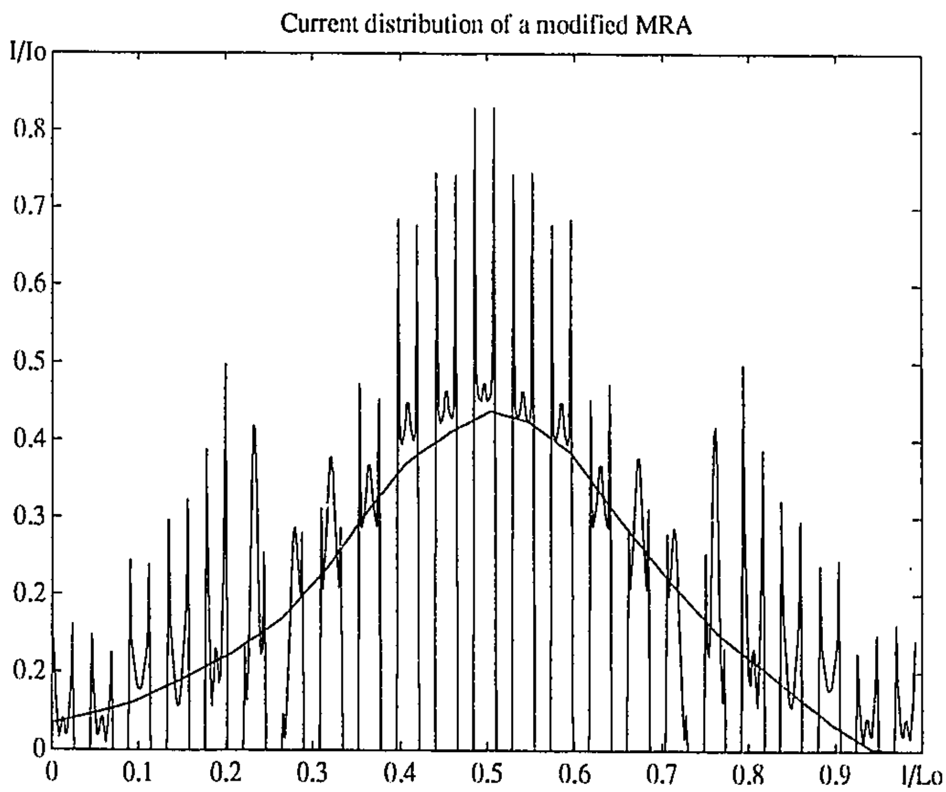


Figure 4.54: The current distribution over a cross-section of the modified MRA. The axis L/Lo is the relative position of the sampling point and I/Io is the relative magnitude of the sampled current value.

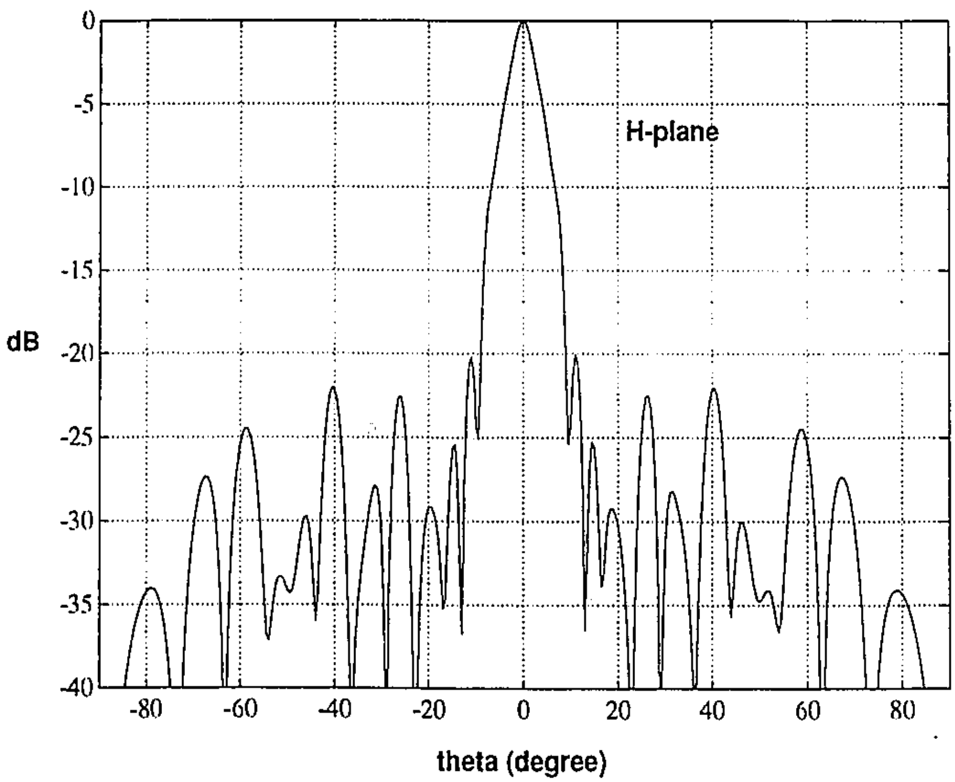


Figure 4.55: The H-plane radiation pattern of the modified MRA.

In our modeling studies, we have simulated some very large MRAs. Here, we give one example. Fig. 4.56 depicts the layout of a microstrip reflectarray. The layout was generated by an automatic mesh generator which was specially designed for our algorithm. This array has been designed to operate at a frequency of 10GHz. The substrate used has a dielectric constant $\epsilon_r = 2.32$ and thickness $h = 2.36mm$. The size of the array is $508.3 \times 394.5mm^2$, and has a total of 914 elements. The main beam of this array is designed to point to $(\theta, \phi) = (30^0, 0^0)$. The feed antenna is positioned so as to be symmetrically located with respect to the four edges of the array, as well being separated from the array, in the third dimension, buy a distance of $1000mm$. The main beam of the feed antenna is directed to illuminate the center of the MRA.

A three-dimensional view of the current distribution on a part of the MRA is depicted in Fig. 4.57. To show a overall view of the current distribution, a color map of the the magnitude and a contour map of the phase of the current distribution on the array are shown in Figs.4.58 and 4.59, respectively. From these two figures, it can be seen that the current distribution exhibits a well defined pattern for both its phase and magnitude. Comparing Figs.4.58, 4.59 and Fig. 4.56, one can see that there exists a clear relationship between the pattern of the current distribution and the distribution of the tail length across the array. We can say that it is this pattern, combined with the radiation pattern of the feed antenna, that generates the coherent phase front of the re-radiated field from the array in the desired direction.

Based on the current distribution, the MRA's re-radiated field is calculated. The radiation patterns are shown in Figs.4.60 and 4.61. The pattern has symmetric side lobes in the H-plane and asymmetric side lobes in the E-plane due to the symmetry and asymmetry of the MRA's layout in these two planes.

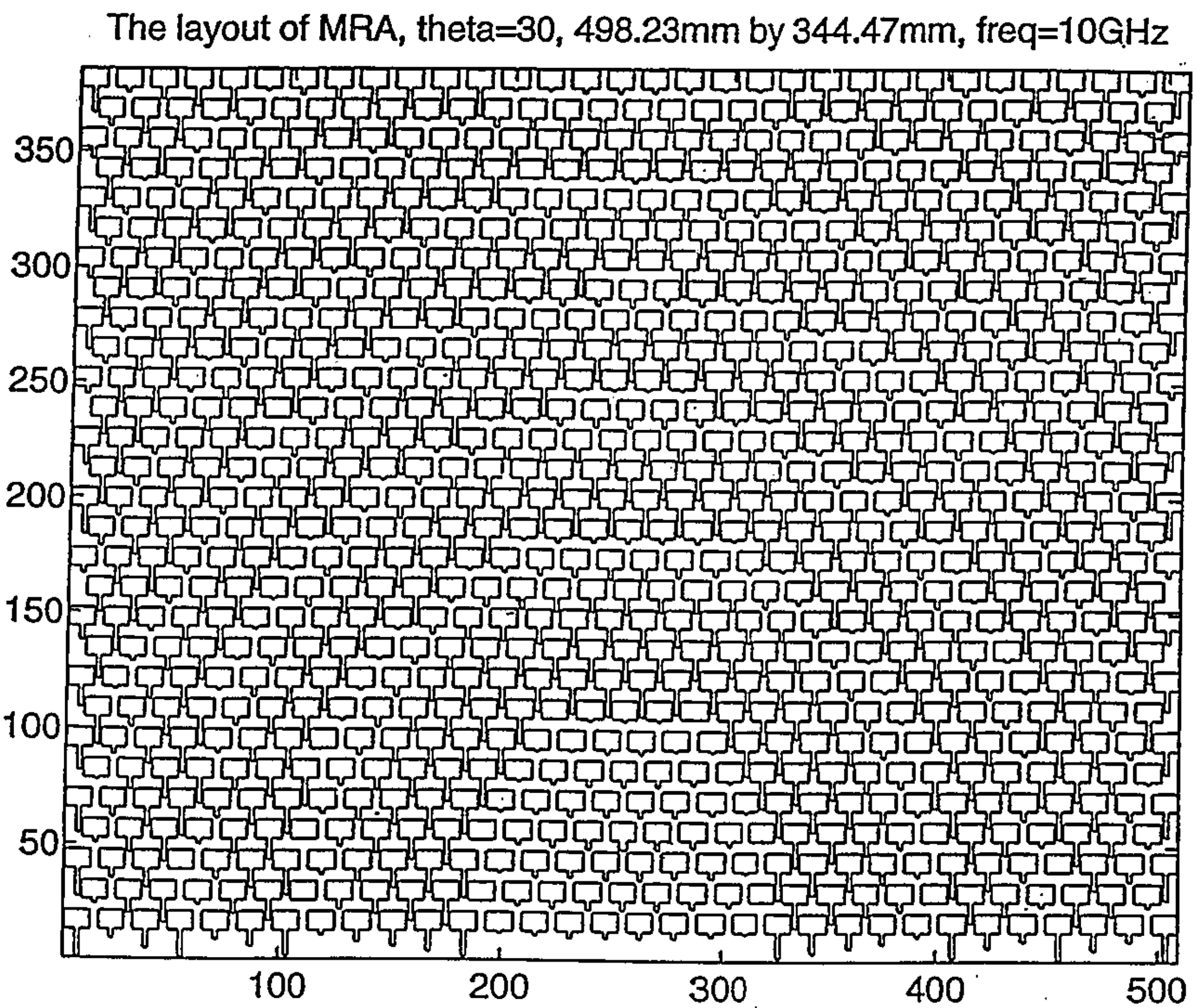


Figure 4.56: The contour-map of the layout of the 32 × 29 MRA, generated by an automatic mesh generator. The vertical axis is x-axis and horizontal axis is y-axis

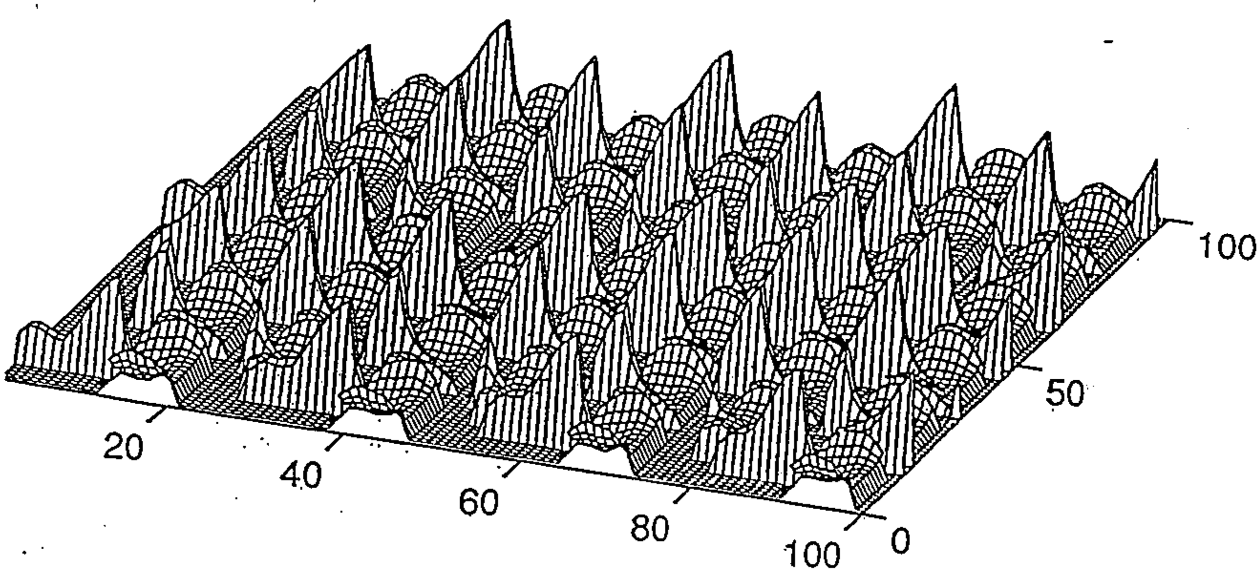


Figure 4.57: A three-dimensional view of the co-polar current distribution on a portion of the 32 × 29 MRA.

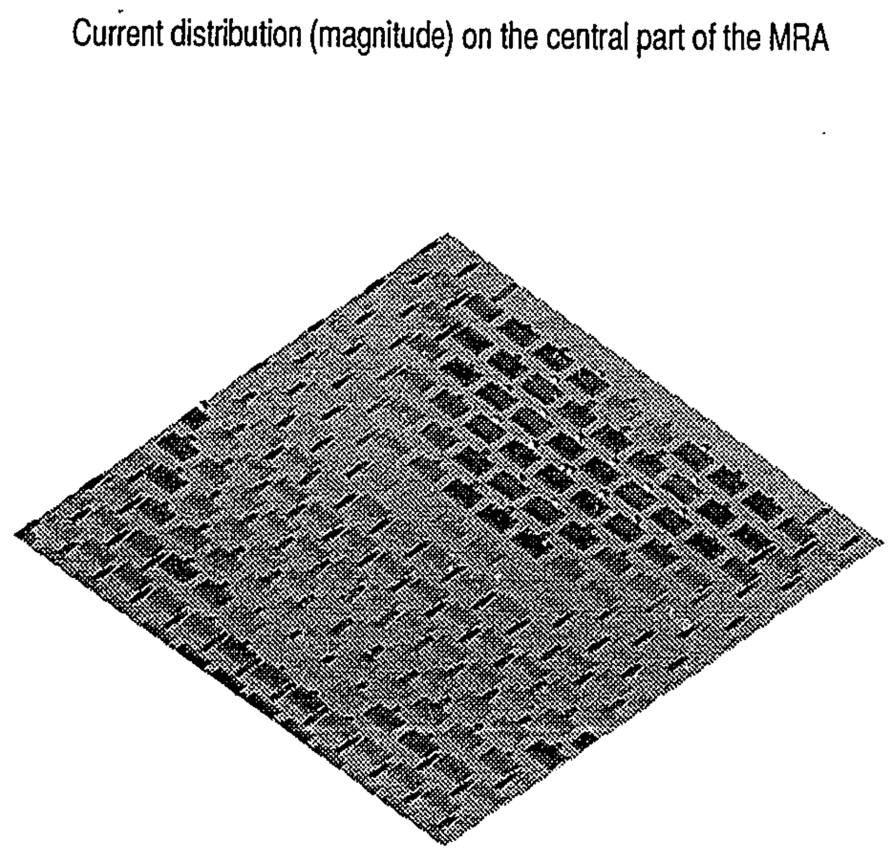


Figure 4.58: Color map of the current magnitude distribution on a 32×29 MRA. The denser color represents the larger magnitude.

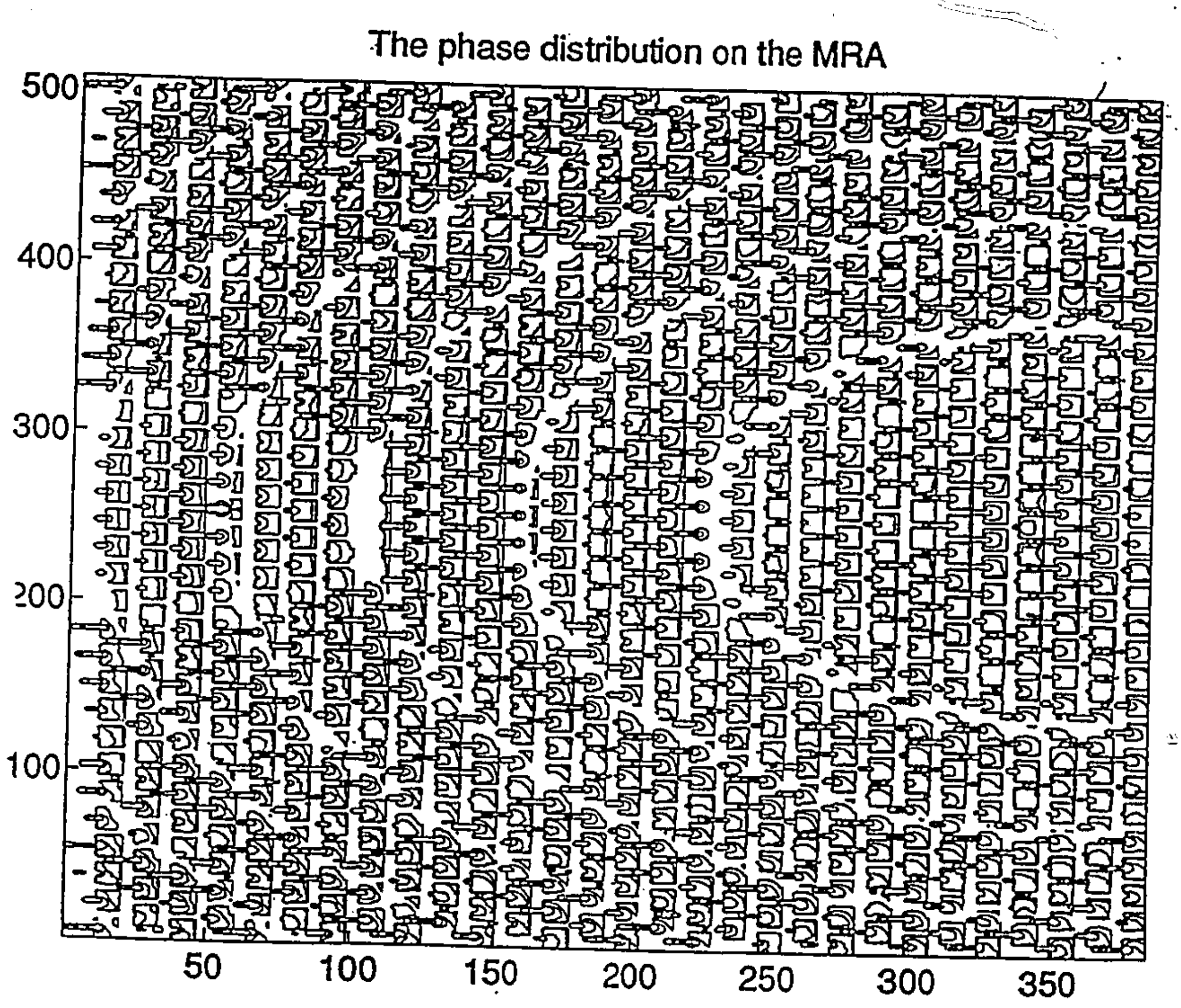


Figure 4.59: Contour map of the current phase distribution on a 32 × 29 MRA.

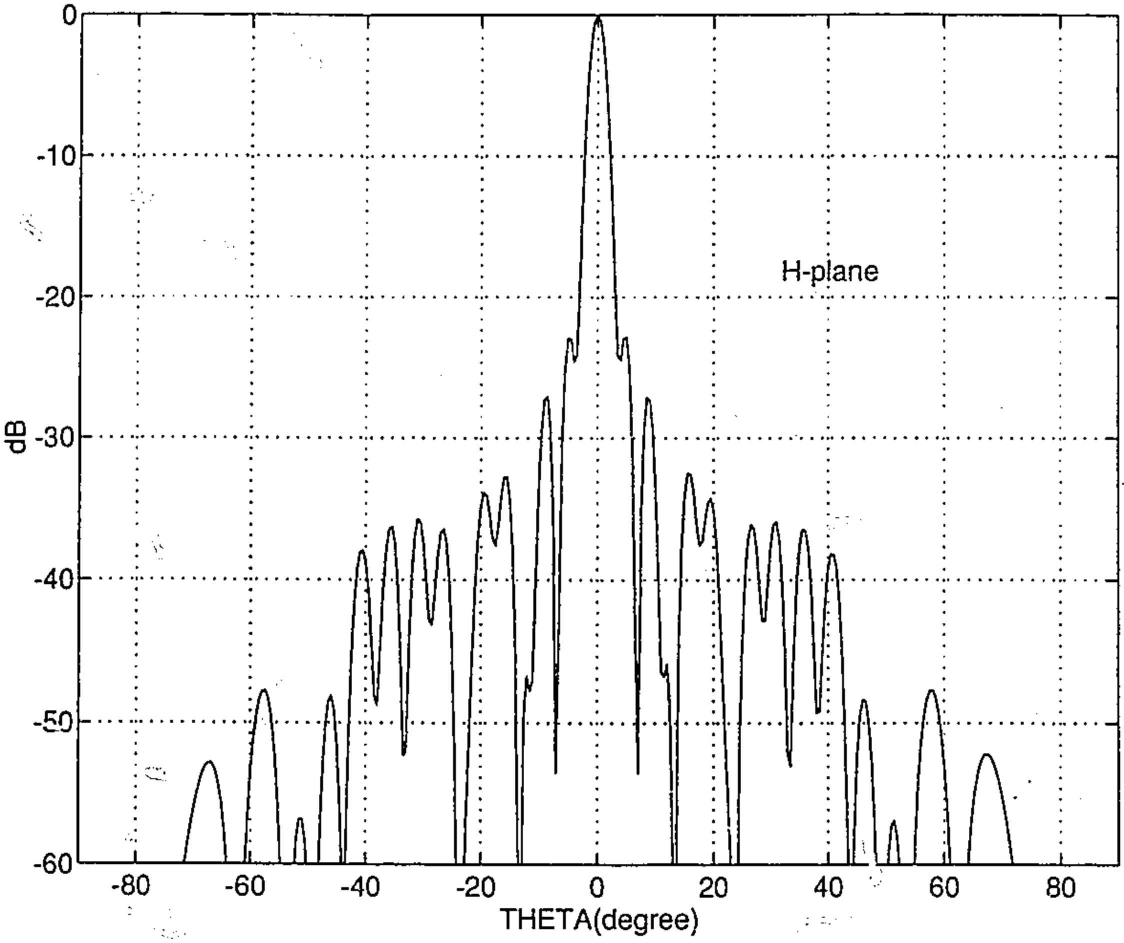


Figure 4.60: The simulated H-plane pattern of a 32×29 MRA.

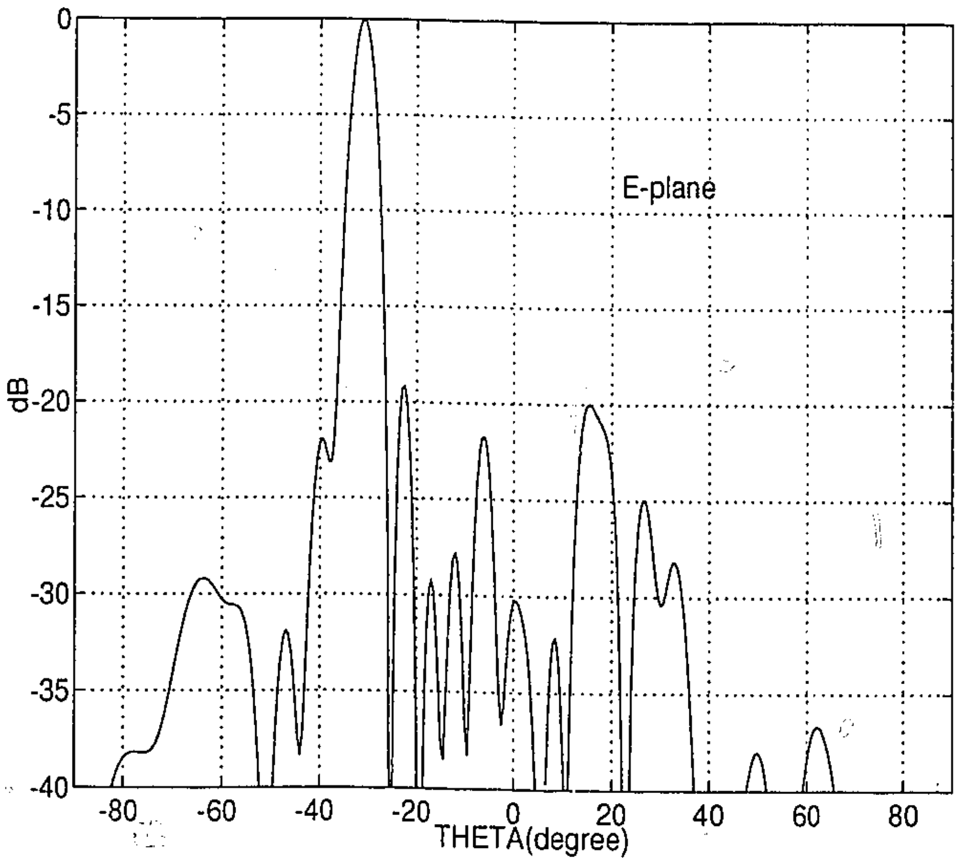


Figure 4.61: The simulated E-plane pattern of a 32×29 MRA.

4.4.4 Discussion

Based on our modeling and experimental studies of MRAs, we make the following conclusions. In order to achieve accurate focusing of the microstrip reflectarray, one must be able to determine the precise phase shift of the re-radiated signal for each of the patches in the array. This phase is controlled by the length of the attached short transmission lines. Intuitively, a linear relationship is expected between the phase shift and the length of the tails. This can be proven true if the equivalent transmission line model is used. But at higher frequencies (e.g. X-band) and in an array environment, the tail cannot simply be treated as a pure transmission line. The effects of re-radiation, surface waves and mutual coupling between neighboring patches and tails, etc., have to be considered. This is true in the analysis of the complete array, as well as for predicting the phase change imposed at each patch. The simulation results in this thesis show that phase shift vs. tail length is not a simple linear relationship. In order to achieve high performance, the destructive edge and tail effects need to be considered. By putting some dummy elements at the edge of the array and by using two tuning approaches, we can improve the performance of the MRA. The simulation results provide corroboration for this idea. Continuing experimental research is needed to further improve the design of microstrip reflectarrays.

The modeling studies that have been carried out in this thesis on a number of large MRAs serve to demonstrate the power of the hybrid CG-FFT for analyzing very large microstrip structures. It is worth noting that that the number of the unknowns that were used to simulate the MRA in Fig. 4.56 is approximately given by $400 \times 500 \times 2 = 400,000$. In the case of the Method of Moments (MoM), where a direct inversion scheme is used, this number grows astronomically. With MoM, the size of the matrix that must be processed in order to carry out this simulation would become $400,000 \times 400,000$. This would require a computer with at least 320G Bytes of memory, which greatly exceeds the capability of even the largest computers.

4.5 Summary

In this chapter, the hybrid CG-FFT technique has been used to simulate several types of microstrip antenna arrays, which include microstrip dipole arrays, microstrip planar arrays and microstrip reflectarrays. Some issues of technical interests for microstrip array design have been studied. In the study, it has been found that

- the current distributions, input impedances of the array elements and the array radiation patterns are all affected by the sizes and shapes of the array, as well as the location of the edges. Due to the existence of surface waves within a microstrip structure, the size, shape and edge factors have a much stronger effect on a microstrip array than on an ordinary array in free space.
- to design large microstrip planar arrays with microstrip line feed networks, serious attention has to be paid to the spurious radiation from the feed network. In order to achieve good pattern synthesis, accurate predictions must be made of spurious radiation.

The modeling studies that have been carried on microstrip reflectarrays provide a good demonstration of the strength of our technique for analyzing complicated and large microstrip structures. A design scheme has been proposed based on the hybrid CG-FFT technique. Some design examples have been tried numerically and experimentally. A proposal for improving the MRA's performance is presented and numerically demonstrated. Further research work needs to be continued.

From the studies in this chapter, we conclude that

- for the design of microstrip arrays with high performance, the finite array model has significant advantages over the infinite array model; and
- the proposed full-wave hybrid CG-FFT method is a very accurate and efficient technique for accomplishing this task.

Part II

The Spectrally Preconditioned CG-FFT Method

Chapter 5

The Development of the Spectrally Preconditioned CG-FFT Method

5.1 Introduction

It has been demonstrated that the CG-FFT method has great advantages over the standard Method of Moments (MoM) for carrying out EM analysis of large scale problems. This is because, as an iterative algorithm, the CG method can reduce the computer memory from $O(N_s^2)$, which is required by the MoM, to $O(N_s)$. At the same time, the integral equation which describes the EM problems of interest to us possesses a convolution kernel. This means that it can be efficiently computed by using the fast Fourier transform (FFT), with the computing time proportional to $O(N_i N_s \log N_s)$ in contrast to the $O(N_s^3)$ computations required by the MoM.

Several different implementation schemes, as described in chapter 2, have been developed in order to enhance the performance of the CG-FFT method. However, in all cases the derivations for these enhancements have had electromagnetisc based starting point. Examples of these enhancements are provided by the treatment of the grad-div operator and the application of circular convolution to carry out the operator calculations. Actually, there is another path which has great potential for improving the performance of existing CG-FFT methods, that is improvements to the CG algorithm itself.

A so-called *biconjugate gradient (BiCG) method* has been developed by Lanczos [70] [71]. It has recently been successfully applied to MoM problems [72] [73]. In the BiCG method, the biorthogonality condition

$$< \overline{r}_{n+1}, r_n >=< r_{n+1}, \overline{r}_n >= 0 \tag{5.95}$$

and biconjugacy condition

$$< \overline{p}_{n+1}, AP_n >=< P_{n+1}, A^a\overline{p}_n >= 0 \tag{5.96}$$

are used in the development of the algorithm. These are different from the condition that is used in the CG algorithm (see section 2.2). It has been shown that the BiCG method is better suited than the CG algorithm for solving problems that are poorly conditioned. In section 2.2, the CG algorithm is applied to a normal equation whose condition number is the square of the system matrix itself. However, the BiCG algorithm has a potential flaw [73]. The condition $< \overline{r}_n, r_n >= 0$ can occur even when $||\overline{r}_n|| \neq 0, \quad ||r_n|| \neq 0$. Under this condition, the BiCG stagnates and fails to yield a solution. In addition, the termination of the BiCG routine is much more complicated because the residuals produced at each step do not have a monotonically decreasing norm.

Another method that has been developed is the *generalized conjugate residual (GCR) method* [74][75]. It has been shown that for appropriate discretization of the EFIE, the GCR method has better convergence than the CG method. But this method has good performance only when the Hermitian part of the system matrix is positive definite. Another disadvantage of GCR is that after the pth iteration, all p of the previous search directions must be saved. This increases the computer memory requirement.

Researchers in applied mathematics have found that in terms of convergence, iterative algorithms work best when applied to linear systems whose iterative matrices have eigenvalue spectra constrained to a small region in the complex plane or on a small interval of the positive axis [76]. A preconditioned conjugate gradient method was developed based on this observation [77] [78]. Preconditioning can be used to reduce the number of iterations necessary to yield a solution of desired accuracy. This is achieved by transferring the equation to an equivalent one with eigenvalues in a more favorable location, i.e., in a small cluster. However, it is not an easy task to implement this method, because it is very difficult to choose a proper preconditioner based solely on observations. Preconditioning can be considered more of an art than a science. Therefore, even though the technique has been applied successfully in other fields of engineering [79], the use of preconditioning in the electromagnetic analysis is not very popular. Actually, this method has great potential in the area of EM analysis. In [80], coaxial collinear (COCO) antenna arrays were analyzed by means of the preconditioned complex conjugate gradient (PCCG) method and very positive results were obtained. The incomplete LU decomposition preconditioning is adopted in this approach. Compared with the standard complex conjugate gradient method (CCG), the technique used in [80] requires much more computer memory for its preconditioning process, in addition the technique uses twice the computing time of the standard CCG method in each CG recursive iteration. In the analysis, the authors have not taken the advantage of the FFT to calculate the convolution integral. Therefore this approach is not efficient in terms both of computer memory and computing time. Another published result describing the application of the PCG method to EM analysis was given in [81]. The authors briefly described the approach used for preconditioning which consisted of using $(A+U)^{-1}$ to multiply both sides of the linear system equation. Unfortunately, the authors did not give many details on the procedure used in implementing their approach. Some details for the use of the PCG method are given in [82]. The Strang preconditioner was duplicated by Smith [82] and applied to EM analysis. The studies carried by Smith [82] showed that this method was effective for one-dimensional problems. However, they failed to produce any positive results for the application of this technique to two-dimensional structures.

In this chapter, a highly efficient spectrally preconditioned CG-FFT (SPCG-FFT) method is presented. This method takes full advantage of the FFT by carrying out the CG operation, constructing the preconditioner and applying the preconditioner all in the spectral domain. It has the advantage that no additional memory and only $O(NlogN)$ additional operations are needed for storing and constructing the preconditioner and performing the preconditioning. At the same time, it has superior convergence properties as compared with any other CG-FFT schemes. In this chapter, we will first introduce the concept of preconditioning and the existing approaches for constructing the preconditioners. Following the introduction, the method for designing spectral preconditioners is described by referring to linear system theory. An outline is presented for the *Toeplitz* preconditioners developed by Strang and Chan. Then a systematic approach of constructing a class of preconditioners is presented. The proposed approach is then extended to treat non-*Toeplitz* matrices that arise in different EM problems. The analysis of the spectrum, convergence performance and numerical experiments will be presented in the next chapter.

5.2 The Concept of Preconditioning

When using the Conjugate Gradient (CG) method for solving a Hermitian positive definite (HPD) linear system Ax=b, one minimizes the quadratic functional

$$\phi(x) = \frac{1}{2}x^T Ax - x^T b \tag{5.97}$$

with the unique minimum giving the desired solution. The convergence rate of the CG method depends on the spectrum of A. Generally speaking, the CG method converges faster if A has a number of clustered eigenvalues. In order to accelerate its convergence rate, a preconditioning step can be introduced at each CG iteration, which leads to the Preconditioned Conjugate Gradient (PCG) method [16].

The procedure for preconditioning A is to find a nonsingular symmetric matrix C, such that $\tilde{A} = C^{-1}AC^{-1}$, and $\tilde{A}$ has an improved condition number. Then the CG algorithm can be applied to the transferred system

$$\tilde{A}\tilde{x} = \tilde{b} \tag{5.98}$$

where $x = C^{-1}\tilde{x}$ and $\tilde{b} = C^{-1}b$.

In detail, the procedure is given by the following:

$$\tilde{x}_0 = 0 \quad , \quad \tilde{r}_o = \tilde{b} \tag{5.99}$$

$$For \quad k = 1, ..., n \tag{5.100}$$

$$if \quad \tilde{r}_{k-1} = 0 \;\; then \tag{5.101}$$

$$set \;\; \tilde{x} = \tilde{x}_{k-1} \quad and \;\; quit \tag{5.102}$$

$$else \quad \beta_k = \frac{< \tilde{r}_{k-1}, \tilde{r}_{k-1} >}{< \tilde{r}_{k-2}, \tilde{r}_{k-2} >} \qquad (\beta_1 \equiv 0) \tag{5.103}$$

$$\tilde{p}_k = \tilde{r}_k + \beta_k \tilde{p}_{k-1} \qquad (\tilde{p}_1 \equiv \tilde{r}_0) \tag{5.104}$$

$$\alpha_k = \frac{< \tilde{r}_{k-1}, \tilde{r}_{k-1} >}{< C^{-1}\tilde{p}_k, AC^{-1}\tilde{p}_k >} \tag{5.105}$$

$$\tilde{x}_k = \tilde{x}_{k-1} + \alpha_k \tilde{p}_k \tag{5.106}$$

$$\tilde{r}_k = \tilde{r}_{k-1} - \alpha_k C^{-1} A C^{-1} \tilde{p}_k \tag{5.107}$$

$$\tilde{x}_k = \tilde{x}_n. \tag{5.108}$$

Here $\tilde{x}_k$ can be regarded as approximations to $\tilde{x}$. If we define

$$P = C^2, \quad p_k = C^{-1}\tilde{p}_k, \quad x_k = C^{-1}\tilde{x}_k, \quad q_k = C^{-1}\tilde{r}_k = b - Ax_k,$$

then the above procedure becomes,

$$x_0 = 0 \quad , \quad r_0 = b \tag{5.109}$$

$$For \quad k = 1, \ldots n \tag{5.110}$$

$$if \quad r_{k-1} = 0 \quad then \tag{5.111}$$

$$set \quad x = x_{k-1} \quad and \quad quit \tag{5.112}$$

$$else \quad solve \quad Pq_{k-1} = r_{k-1} \quad for \quad q_{k-1} \tag{5.113}$$

$$\beta_k = \frac{< q_{k-1}, r_{k-1} >}{< q_{k-2}, r_{k-2} >} \tag{5.114}$$

$$p_k = q_k + \beta_k r_{k-1} \tag{5.115}$$

$$\alpha_k = \frac{< q_{k-1}, r_{k-1} >}{< p_k, Ap_k >} \tag{5.116}$$

$$x_k = x_{k-1} + \alpha_k p_k \tag{5.117}$$

$$r_k = r_{k-1} - \alpha_k A p_k \tag{5.118}$$

$$x_k = x_n. \tag{5.119}$$

The above procedure is called the *Preconditioned Conjugate Gradient* algorithm, and the matrix P is called the *preconditioner*. It can be shown that when the CG algorithm is generated in this way, the residuals and the search directions satisfy

$$< r_j, P^{-1} r_i >= 0 \qquad < C^{-1} p_j, AC^{-1} P_i >= 0. \tag{5.120}$$

The above preconditioning procedure is for HPD systems, which means that matrix A must be Hermitian positive definite (HPD). However, the system matrices which arise in the case of electromagnetic radiation and scattering problems cannot be guaranteed to be HPD. To extend this preconditioning concept to those problems where the matrices are not HPD, the biconjugate gradient can be used as the basic algorithm for applying preconditioning. In this algorithm, the matrix A need not be HPD, but if it is, the algorithm reduces to the standard CG algorithm. In our EM analysis, the system matrix A is complex symmetric when the structure is properly discretized. Under this condition, we give the BiCG algorithm below

$$x_0 = 0 \quad , \quad r_0 = b \tag{5.121}$$

$$For \quad k = 1, ...n \tag{5.122}$$

$$if \quad r_{k-1} = 0 \;\; then \tag{5.123}$$

$$set \;\; x = x_{k-1} \quad and \;\; quit \tag{5.124}$$

$$else \quad \beta_k = \frac{< r_{k-1}^*, r_{k-1} >}{< r_{k-2}^*, r_{k-2} >} \tag{5.125}$$

$$p_k = r_k + \beta_k p_{k-1} \tag{5.126}$$

$$\alpha_k = \frac{< r_{k-1}^*, r_{k-1} >}{< p_k^*, Ap_k >} \tag{5.127}$$

$$x_k = x_{k-1} + \alpha_k p_k \tag{5.128}$$

$$r_k = r_{k-1} - \alpha_k Ap_k \tag{5.129}$$

$$x_k = x_n. \tag{5.130}$$

After applying a preconditioning step, a preconditioned BiCG algorithm

(PBiCG) can be obtained as

$$x_0 = 0 \quad , \quad r_0 = b \tag{5.131}$$

$$For \quad k = 1, ...n \tag{5.132}$$

$$if \quad r_{k-1} = 0 \;\; then \tag{5.133}$$

$$set \;\; x = x_{k-1} \quad and \;\; quit \tag{5.134}$$

$$else \quad solve \;\; Pq_{k-1} = r_{k-1} \quad for \;\; q_{k-1} \tag{5.135}$$

$$\beta_k = \frac{< r^*_{k-1}, q_{k-1} >}{< r^*_{k-2}, q_{k-2} >} \tag{5.136}$$

$$p_k = q_k + \beta_k p_{k-1} \tag{5.137}$$

$$\alpha_k = \frac{< r^*_{k-1}, q_{k-1} >}{< p^*_k, Ap_k >} \tag{5.138}$$

$$x_k = x_{k-1} + \alpha_k p_k \tag{5.139}$$

$$r_k = r_{k-1} - \alpha_k A p_k \tag{5.140}$$

$$x_k = x_n. \tag{5.141}$$

We may derive another preconditioned CG algorithm based on the standard CG algorithm for normal equations (PCGNR),

$$A^* A x = A^* b, \tag{5.142}$$

since A^*A is always Hermitian no matter if A is Hermitian or not. Actually, the CG algorithm listed in (2.14) is equivalent to the CG algorithm that is used with the normal equations. The CGNR algorithm has some merits, such as guaranteed convergence as the new system matrix A^*A is guaranteed to be an HPD matrix. On the other hand, the CGNR has the shortcoming of slower convergence compared with that of the BiCG. Since the condition number of A^*A is the square of that of A, it follows that when A is ill-conditioned that the convergence of CGNR is often unacceptably slow. The preconditioning process is aimed at improving the condition of the system matrix. With this objective in mind, the BiCG algorithm is used in the design and analysis of the preconditioner. The study of preconditioning for the CGNR algorithm can be carried out following the same guidelines.

From the above discussion, it is clear that in order to apply the PCG algorithm effectively, one must have an efficient means of solving the linear system $Pq = r$. Further, a good preconditioner can lead to an algorithm that has very rapid convergence. Therefore, the key issues related to the implementation of the PCG technique are; (i) how to choose P and (ii) how to solve the linear equation $Pq = r$ for q.

It has been proved [16] that a good preconditioner for A is a matrix P which approximates A well ($P^{-1} \approx A^{-1}$) in the sense that the spectrum of the preconditioned matrix $P^{-1}A$ is clustered around 1 or has a small condition number. The convergence of the PCG method on the other hand depends on the distribution of all the eigenvalues of $P^{-1}A$, not exclusively on the external eigenvalues.

In summary, a good preconditioner, P, of matrix A should satisfy the following two criteria:

- P approximates A well in the sense that $P^{-1}A$ has a small condition number or that the spectrum of $P^{-1}A$ has a certain clustering feature.

- P can be inverted effectively or the linear equation $Pq = r$ can be solved easily.

5.3 The Design of Preconditioners

An approach that can be used to construct a preconditioner involves computing an *incomplete Cholesky factorization* of A. The idea behind this approach is to find a lower triangular matrix L with the property that L approximates the A's exact Cholesky factor T. The preconditioner is then taken as $P = LL^T$. The disadvantage of this approach is that the matrix and its incomplete Cholesky factor L have to be stored. This would be suitable for the solution of a system with a very sparse system matrix. However, in EM analysis, the system matrix (impedance matrix for the integral equation method) is always a full, dense matrix. If the incomplete Cholesky factorization approach for preconditioning is employed in the PCG algorithm, it is found that the savings in memory, which one would expect when implementing the CG algorithm, is not realized. Thus, this preconditioning approach is not a promising one in the case of EM analysis.

There have been some other types of preconditioners developed. Such as *incomplete block preconditioners*, which employ the incomplete block Cholesky factorization, and *domain decomposition preconditioning* for the solution of elliptic partial differential equations. Because they are developed for different systems, with system matrices having special properties, the amount of extra computer memory and computing time required for all these approaches is closely associated with the special structures of the system matrices. They are generally not suited to electromagnetic problems.

In EM analysis, the impedance matrix has special features, which are the results of the convolution property of the integral equations. For example, for one-dimensional EM problems, the impedance matrices are symmetric *Toeplitz* matrices. Following discretization, it is found that in the case of other types of

EM problems, the system matrices are also related to *Toeplitz* matrices in some way. *Toeplitz*-type matrices have a number of features that can be used when developing a preconditioner. The first *Toeplitz* preconditioner was proposed by Strang. The Strang preconditioner S is obtained by preserving the central half diagonal of matrix A and using it to form a circulant matrix (see Section 5.5.2 for details). Since S is circulant, the matrix-vector product $S^{-1}r$ can be conveniently calculated via the FFT with $O(NlogN)$ operations. In the Strang preconditioner, only half of the elements of A are used. It is suggested that a better preconditioner can be obtained if all of the elements of A can be used.

In this thesis, we present another approach for constructing a class of preconditioners. This approach is stimulated by the Strang *Toeplitz* preconditioner and developed using linear system theory. All of the elements of A are used in our approach and at the same time the preconditioning is carried out in the frequency domain by way of the FFT. It is shown that this approach has a better convergence rate compared with Strang's approach while at the same time being as computationally efficient as the Strang preconditioner. The Strang preconditioner only deals with the SPD *Toeplitz* system. In this thesis we extend our approach to design preconditioners for other types of systems, such as Block-*Toeplitz* - *Toeplitz* -Matrix (BBT-Matrix) systems and block-BBT systems.

Since the design scheme of the spectral preconditioner in this thesis originates in the application of circular convolution and linear convolution of linear system theory, it is instructive to start with the introduction of this application.

5.4 Linear and Circular Discrete Convolution

In this section we show that a linear equation composed of a circular matrix can be solved accurately using circular deconvolution via the FFT while a linear equation composed of a *Toeplitz* matrix can only be solved approximately by the linear deconvolution via an FFT.

In linear system theory [83], any linear time-invariant system can be represented by its impulse response. That is, if the input discrete sequence is

$$\Theta(\kappa) = \begin{cases} \theta_\kappa & 0 \le \kappa \le N-1 \\ 0 & otherwise \end{cases} \tag{5.143}$$

and the impulse response of the system is

$$\Psi(\kappa) = \begin{cases} \psi_{|\kappa|} & -(N-1) \le \kappa \le N-1 \\ 0 & otherwise \end{cases}, \tag{5.144}$$

the output discrete sequence β can be obtained by carrying out the linear convolution of Θ and Ψ

$$\beta(n) = \Psi(n) * \Theta(n) = \sum_{m=-\infty}^{\infty} \Psi(n-m)\Theta(m) = \sum_{m=-\infty}^{\infty} \psi_{|n-m|}\theta_m, \tag{5.145}$$

where β is an infinite sequence with duration $3N-2$.

Similarly, if the impulse response of a linear system is an N-periodic sequence

$$\hat{\Xi}(n) = \xi_N(n[\text{mod } N]) \tag{5.146}$$

$$\xi_N = \{\xi_0, \xi_1, ..., \xi_{N-1}\}, \tag{5.147}$$

for an N-periodic input sequence

$$\hat{\Theta}(n) = \vartheta_N(n[\text{mod } N]) \tag{5.148}$$

$$\vartheta_N = \{\vartheta_0, \vartheta_1, ..., \vartheta_{N-1}\}, \tag{5.149}$$

the output sequence of the system can be obtained using the circular convolution

$$\hat{\beta}(n) = \hat{\Xi}(n) \otimes \hat{\Theta}(n) = \sum_{m=-\infty}^{\infty} \hat{\Xi}(n-m)\hat{\Theta}(m) \tag{5.150}$$

$$= \sum_{m=-\infty}^{\infty} \xi_N((n-m)[\text{mod } N\,])\vartheta_N(n[\text{mod } N]) \tag{5.151}$$

and $\hat{\beta}$ is also an N-periodic sequence whose periods are given by

$$\beta_N = \{\beta_0, \beta_1, ..., \beta_{N-1}\}. \tag{5.152}$$

Now, we define an $N \times N$ Toeplitz matrix Ψ_N,i.e.,

$$\Psi_N = \begin{pmatrix} \psi_0 & \psi_1 & \psi_2 & \dots & \psi_{N-1} \\ \psi_1 & \psi_0 & \psi_1 & \dots & \psi_{N-2} \\ \dots & \dots & \dots & \dots & \dots \\ \psi_{N-2} & \psi_{N-3} & \dots & \psi_2 & \psi_1 \\ \psi_{N-1} & \psi_{N-2} & \dots & \psi_1 & \psi_0 \end{pmatrix} \quad \Psi_N(i,j) = \psi_{|i-j|}, \tag{5.153}$$

an $N \times N$ circular matrix Ξ_N, i.e.,

$$\Xi_N = \begin{pmatrix} \xi_0 & \xi_1 & \dots & \xi_{N-2} & \xi_{N-1} \\ \xi_{N-1} & \xi_0 & \dots & \xi_{N-3} & \xi_{N-2} \\ \dots & \dots & \dots & \dots & \dots \\ \xi_2 & \xi_3 & \dots & \xi_0 & \xi_1 \\ \xi_1 & \xi_2 & \dots & \xi_{N-1} & \xi_0 \end{pmatrix} \quad \Xi_N(i,j) = \xi_{(i-j)\text{mod } N} \tag{5.154}$$

and an arbitrary vector

$$\vartheta_N = \{\vartheta_0, \vartheta_1, ..., \vartheta_{N-1}\}^T. \tag{5.155}$$

The elements of the *Toeplitz* matrix-vector product can be written as

$$\Phi(n) = \Psi_N(n)\vartheta_N(n) = \sum_{m=0}^{N-1} \psi_{|n-m|}\vartheta_m \quad 0 \le n \le N-1. \tag{5.156}$$

Similarly, the elements of the circular matrix-vector product can be written as

$$\Phi_N(n) = \Xi_N(n)\vartheta_N(n) = \sum_{m=0}^{N-1} \xi_{(n-m) \bmod N}\vartheta_m \quad 0 \leq n \leq N-1. \tag{5.157}$$

Comparing 5.156 with 5.145, and 5.157 with 5.151, we can find that the Toeplitz matrix vector production and circular matrix-vector production are embedded in the linear convolution and circular convolution, respectively. That is

$$\begin{aligned} \Phi(n) &= \beta(n) && for\ \ 0 \leq n \leq N-1, \quad and \\ \Phi_N(n) &= \hat{\beta}(n) = \beta_N(n) && for\ \ 0 \leq n \leq N-1. \end{aligned} \tag{5.158}$$

Therefore, solving the linear equations,

$$\Psi_N\vartheta_N = \Phi, \quad and \tag{5.159}$$

$$\Xi_N\vartheta_N = \Phi_N, \tag{5.160}$$

is equivalent to determining the input signal sequences $\Theta, \hat{\Theta}$ based on the output $\beta, \hat{\beta}$ and the impulse response $\Psi, \hat{\Xi}$ of a linear system.

According to the convolution theorem [83], the circular convolution and deconvolution can be carried out efficiently via the FFT. This means that if we apply the DFT to the periodic sequences $\hat{\Theta}, \hat{\beta}$ and $\hat{\Xi}$, i.e., to both side of 5.151, we have

$$\tilde{\hat{\beta}} = \tilde{\hat{\Theta}}\tilde{\hat{\Xi}}, \quad or \quad \tilde{\hat{\Theta}} = \frac{\tilde{\hat{\beta}}}{\tilde{\hat{\Xi}}}. \tag{5.161}$$

Therefore the circular convolution or the circular matrix-vector product $\Xi_N\vartheta_N$ can be obtained as

$$\hat{\beta} = \mathcal{F}^{-1}\{\tilde{\hat{\Theta}}\tilde{\hat{\Xi}}\} \tag{5.162}$$

with $O(NlogN)$ operations. Similarly, the circular deconvolution and the solution of the linear equation $\Xi_N\vartheta_N = \hat{\beta}$ can also be obtained as

$$\hat{\Theta} = \mathcal{F}^{-1}\{\frac{\tilde{\hat{\beta}}}{\tilde{\hat{\Xi}}}\} \tag{5.163}$$

with the same number of operations.

It is also possible to compute the linear convolution (equivalently the *Toeplitz* matrix-vector production) and correspondingly linear deconvolution with the FFT. But one has to be very careful when the FFT is used to solve the linear equation $\Psi_N \vartheta_N = \Phi_N$.

To perform the linear convolution, we may view β, Θ, Ψ as if they were all $(3N-2)$-periodic sequences, and treat the linear convolution 5.145 as a $(3N-2)$-point circular convolution. It is because $\Psi_N \vartheta_N = \Phi_N$ can be embedded in the linear convolution and that all nontrivial $(2N-1)$ values of Ψ and N values of Θ are known, that we can calculate β as well as Φ_N without any errors by following the same procedure 5.158 - 5.160.

However, it is not easy to solve the equation $\Psi_N \vartheta_N = \Phi$, or say to perform $\vartheta_N = \Psi_N^{-1}\Phi$. This is because we only have N values for the output β, that is Φ, therefore there isn't sufficient information to carry out the linear deconvolution. Thus the product $\vartheta_N = \Psi_N^{-1}\Phi$ only partially characterizes a linear deconvolution. But this partially characterized linear convolution can still be approximated by a circular deconvolution and then calculated using an FFT.

5.5 The Design of Spectral Preconditioners

5.5.1 General Description

In the PCG algorithm, the important step is to solve the preconditioning linear equation

$$P q_k = r_k, \tag{5.164}$$

We know that the preconditioner $P^{-1} \approx A^{-1}$. Therefore solving this equation is approximately equivalent to solving

$$Aq_k = r_k \qquad \Longrightarrow q_k = A^{-1} r_k. \tag{5.165}$$

In EM analysis, the system matrices (impedance matrices) are *Toeplitz* matrices (1D case) or block-*Toeplitz* matrices (e.g. linear antenna array) or other types of matrices which may relate to *Toeplitz* matrices some way. For example, consider a z-directed microstrip strip (as shown in Fig.5.62) of length L and width w, exposed to an incident field, $E_z^i(z)$, which is parallel to the z-axis. When $w \ll \lambda$, the x-directed current J_x is much smaller than the z-directed current J_z. As well, J_z is approximately evenly distributed in the x direction. Therefore the integral equation describing the induced current and incident field is,

$$\frac{1}{j\omega\epsilon}\int_{-L/2}^{L/2} J_z(z')\left[\frac{\partial^2 G(z-z')}{\partial z^2} + k^2 G(z-z')\right] dz' = -E_z^i(z) \tag{5.166}$$

where G is the Green's function for this microstrip structure. As described in Chapter 2, this integral equation can be transferred to a linear system equation,

$$AJ = V, \tag{5.167}$$

after it is discretized by using basis functions and testing functions. Corresponding to the convolution property of this integral, the system matrix A is a *Toeplitz* matrix.

For an array of the above microstrip lines (as shown in Fig.5.62), the integral equation is

$$\frac{1}{j\omega\epsilon}\sum_{n=0}^{N-1}\int_{-L/2}^{L/2} J_z(z', x_n)\left[\frac{\partial^2 G(z-z', x_n - x_m)}{\partial z^2} + k^2 G(z-z', x_n - x_m)\right] dz'$$
$$= -E_z^i(z, x_m) \qquad m = 0, 1, ..., N-1. \tag{5.168}$$

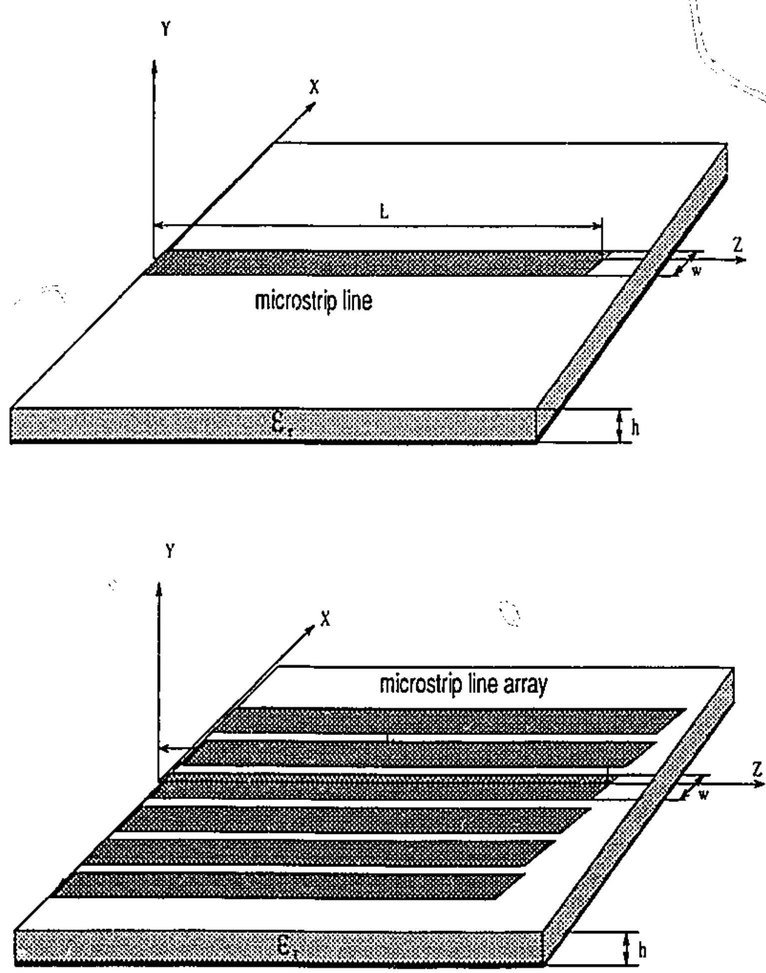


Figure 5.62: The geometry of a long microstrip line

The resulting matrix A is an $N \times N$ block *Toeplitz* matrix with each block being a $\mu \times \mu$ *Toeplitz* matrix (μ is the number of expansion segments on each line). In this thesis we call it as *BTT-matrix.*

For planar microstrip structures, the general integral equations can be written as

$$\frac{1}{j\omega\epsilon} \int\int_s [L_{11}(GJ_x) + L_{12}(GJ_y)]\, ds' = E_x^i \tag{5.169}$$

$$\frac{1}{j\omega\epsilon} \int\int_s [L_{12}(GJ_x) + L_{22}(GJ_y)]\, ds' = E_x^i \tag{5.170}$$

with

$$L_{11} \equiv k_0^2 + \frac{\partial^2}{\partial x^2}, \quad L_{12} \equiv \frac{\partial^2}{\partial x \partial y}, \quad L_{22} \equiv k_0^2 + \frac{\partial^2}{\partial y^2}. \tag{5.171}$$

After dividing a structure into a set of $M \times N$ cells, expanding the current using a pulse function and testing the equation by a point matching scheme, we are ready to find

$$\begin{pmatrix} A_{xx} & A_{xy} \\ A_{xy} & A_{yy} \end{pmatrix} \begin{pmatrix} J_x \\ J_y \end{pmatrix} = j\omega\epsilon \begin{pmatrix} E_x \\ E_y \end{pmatrix}. \tag{5.172}$$

Because of the convolution property of the integral equations, A_{xx}, A_{xy} and A_{yy} are all *BTT-matrices.*

If A is a *Toeplitz* matrix, solving the preconditioning equation 5.164 or 5.165 is equivalent to solving 5.159 which was discussed in the last section. As we know, we cannot solve this type of equation accurately using the circular deconvolution technique. But for preconditioning, one doesn't need an exact solution (of course the greater the accuracy the better). Therefore we can still use the circular deconvolution idea to carry out our preconditioning task. In the following sections, we will make the assertion that there are basically two approaches for approximately solving an N-dimensional *Toeplitz* -type linear equation $A_N q_k = r_K$. They are:

1. One can generate another matrix A_{Nc} from A_N. A_{Nc} will have the same size as A_N but will be generated to be a circulant matrix. The new matrix will approximate A_N sufficiently well that the solution of the linear equation

$$A_{Nc}q_k = r_k \tag{5.173}$$

will approximate the solution of its original equation 5.164. Equation 5.173 can be solved efficiently using circular deconvolution via the FFT.

2. One can extend the *Toeplitz* matrix A_N to a $2N \times 2N$ circulant matrix A_{2N} and also extend r_k to $r_k^{(2N)}$ by padding extra N elements following r_k. The new linear equation

$$A_{2N}q_k^{(2N)} = r_k^{(2N)} \tag{5.174}$$

can be solved by circular deconvolution via the FFT. The q_k is extracted from the solution for $q_k^{(2N)}$ in 5.174. The N extra padded elements of $r_k^{(2N)}$ are artificial and cannot be guaranteed to be the true output of the linear system. Thus the solution extracted from $q_K^{(2N)}$ is an approximate solution of the original equation 5.164.

The accuracy of the solution obtained by using the above two approaches is closely related to the manner in which A_{Nc} was generated (using approach #1) and r_k, q_k were extended (using approach #2). In the following sections, we will fully exploit these approaches in the construction of our preconditioners.

5.5.2 Review of *Toeplitz* Preconditioners

We first introduce preconditioners for a *Toeplitz* matrix that have been developed by mathematicians. These *Toeplitz* preconditioners are constructed using approach #1 as described above.

The Strang Preconditioner

Strang proposed [84] the use of a preconditioned conjugate gradient method with a circulant preconditioner for solving a symmetric positive definite (SPD) *Toeplitz* system $A_N x = b$. Following this approach, a circulant preconditioner $\mathcal{S}_N$ is obtained by copying the central diagonals of A_N and bringing them around to complete the circulant. This method falls into the category of approaches identified by #1 above. It can be formulated as follows.

If N=2m (which means the matrix has even numbers of rows and columns), the elements of A_N are given by

$$A_N(i,j) = a_{|i-j|} \quad 0 \leq i,j < N, \tag{5.175}$$

and the elements of $\mathcal{S}_N$ are

$$\mathcal{S}_N(i,j) = s_{|i-j|} \tag{5.176}$$

with

$$s_\kappa = \begin{cases} a_\kappa & 0 \leq \kappa \leq m \\ a_{N-\kappa} & m \leq \kappa < N. \end{cases} \tag{5.177}$$

The number of operations per iteration will be of order $O(NlogN)$ since a circulant system can be solved efficiently using an FFT. In numerical experiments, it is found that the eigenvalues of $\mathcal{S}_N^{-1}A$ cluster around 1, except for the largest and smallest eigenvalues. The numerical example given below demonstrates this point.

For a *Toeplitz* matrix with the first row of elements $a_\kappa = t^\kappa$, a preconditioner $\mathcal{S}_N$ can be composed following the procedure given in 5.175 - 5.177. It can be proven that there are only five distinct eigenvalues for the matrix $\mathcal{S}_N^{-1}A$. These are

$$\tfrac{1}{1+t}, \ \tfrac{1}{1-t}, \ 1, \ \tfrac{1}{1+t^{\frac{N}{2}}} \ \text{and} \ \tfrac{1}{1-t^{\frac{N}{2}}}. \tag{5.178}$$

Thereafter, for large N and $|t| < 1$, most eigenvalues of $\mathcal{S}_N^{-1}A$ cluster around 1, except for the two extremes, which are $\frac{1}{1+t}$ and $\frac{1}{1-t}$. This behavior implies that if the CG method is applied to solve the preconditioned system, such as the one above, the convergence could be extremely fast.

The Chan Preconditioner

It is clear that the Strang preconditioner $\mathcal{S}_N$ only uses half of the information of A_N, the other half is lost in constructing $\mathcal{S}_N$. It is intuitive for people to consider that if more of the information in A_N is included in $\mathcal{S}_N$, the preconditioner may have a better convergence performance.

Chan has proposed another circulant matrix $\mathcal{C}_N$ [85]. It is obtained by averaging the corresponding diagonals of A_N in the manner that is described in the following formulations. This method falls into the category of approaches identified by approach #1 given in section 5.5.1. Specifically, the elements of $\mathcal{C}_N$ are given by

$$\mathcal{C}_N(i,j) = c_{|i-j|} \tag{5.179}$$

with

$$c_\kappa = \frac{\kappa a_{N-\kappa} + (N-\kappa)a_\kappa}{N}, \quad 0 \le \kappa < N. \tag{5.180}$$

It has been proven that such $\mathcal{C}_N$ minimizes the Frobenius norm $||\mathcal{C}_N - A_N||_F$ and numerical experiments have showed that the spectrum of the preconditioned matrix $\mathcal{C}_N^{-1}A_N$ is also clustered around 1 with the conditioner number of $\mathcal{C}_N^{-1}A_N$ often being smaller than that of $\mathcal{S}_N^{-1}A_N$. However, it has also been shown that the Chan preconditioner does not necessarily improve the convergence rate compared with the Strang preconditioner.

5.5.3 The Proposed Spectral Preconditioners

Strang and Chan preconditioners are obtained by directly processing the A_N matrix to construct the circulant matrices $\mathcal{S}_N, \mathcal{C}_N$. The Chan preconditioner improves the Strang preconditioner by including more information about the original matrix A_N in its preconditioner $\mathcal{C}_N$. Stimulated by Chan's idea, we take a different direction and look for other preconditioners of A_N with all the elements of A_N being effectively used so that a better convergence may be achieved.

The starting point of this approach is the construction of a circulant matrix from the system matrix A_N. To do this, Strang cut A_N into two and used one half of A_N to compose a circulant matrix $\mathcal{S}_N$. Therefore, $\mathcal{S}_N$ has the same dimension as A_N. If we recall the approach we used in Chapter 2 to perform the operation $L[J]$ via the FFT, we can find that we also need to compose a circulant convolution from the original linear convolution form. Instead of cutting the size of the system matrix to achieve the goal, we extend the matrix along with the vectors J and E. Here we use the same idea to start our derivation of the circulant preconditioner. Apparently, this method falls into the category of approaches identified by approach #2 given in section 5.5.1.

5.5.3.1 One-Dimensional Linear Structures - *Toeplitz* Preconditioners

In the electromagnetic analysis of one-dimensional linear structures, such as wire antennas or thin, narrow microstrip lines, etc., the resulting system matrix is a

Toeplitz matrix. We define an $N \times N$ system matrix

$$A_N = \begin{pmatrix} a_0 & a_1 & \dots & a_{N-1} & a_{N-1} \\ a_1 & a_0 & a_1 & \dots & a_{N-2} \\ a_2 & a_1 & a_0 & \dots & a_{N-3} \\ \dots & \dots & \dots & \dots & \dots \\ a_{N-2} & a_{N-3} & \dots & a_0 & a_1 \\ a_{N-1} & a_{N-2} & \dots & a_1 & a_0 \end{pmatrix}. \tag{5.181}$$

In order to generate a circulant matrix from the system matrix, we can extend A_N to a $2N \times 2N$ matrix $\mathcal{Z}$ as follows

$$\mathcal{Z} = \begin{pmatrix} \begin{bmatrix} a_0 & a_1 & a_2 & \dots & a_{N-2} & a_{N-1} \\ a_1 & a_0 & a_1 & \dots & a_{N-3} & a_{N-2} \\ \dots & \dots & \dots & \dots & \dots & \\ a_{N-1} & a_{N-2} & a_{N-3} & \dots & a_1 & a_0 \end{bmatrix} & \begin{bmatrix} 0 & a_{N-1} & a_{N-2} & \dots & a_2 & a_1 \\ a_{N-1} & 0 & a_{N-1} & \dots & a_3 & a_2 \\ \dots & \dots & \dots & \dots & \dots & \\ a_1 & a_2 & a_3 & \dots & a_{N-1} & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & a_{N-1} & a_{N-2} & \dots & a_2 & a_1 \\ a_{N-1} & 0 & a_{N-1} & \dots & a_3 & a_2 \\ \dots & \dots & \dots & \dots & \dots & \\ a_1 & a_2 & a_3 & \dots & a_{N-1} & 0 \end{bmatrix} & \begin{bmatrix} a_0 & a_1 & a_2 & \dots & a_{N-2} & a_{N-1} \\ a_1 & a_0 & a_1 & \dots & a_{N-3} & a_{N-2} \\ \dots & \dots & \dots & \dots & \dots & \\ a_{N-1} & a_{N-2} & a_{N-3} & \dots & a_1 & a_0 \end{bmatrix} \end{pmatrix}. \tag{5.182}$$

It is easy to show that $\mathcal{Z}$ is symmetric circulant. If we use A_N^e to denote the submatrices at the upper-right and lower-right corners of $\mathcal{Z}$, the circulant matrix can be rewritten as

$$\mathcal{Z} = \begin{pmatrix} A_N & A_N^e \\ A_N^e & A_N \end{pmatrix}. \tag{5.183}$$

After defining extended vectors

$$q^e = \begin{pmatrix} q \\ \wp \end{pmatrix}, \quad r^e = \begin{pmatrix} r \\ \mho \end{pmatrix}, \tag{5.184}$$

an augmented preconditioning system is obtained as

$$\mathcal{Z}q^c = r^c \quad \Longrightarrow \quad \begin{pmatrix} A_N & A_N^c \\ A_N^c & A_N \end{pmatrix} \begin{pmatrix} q \\ \wp \end{pmatrix} = \begin{pmatrix} r \\ \mho \end{pmatrix}. \tag{5.185}$$

From the discussion in section 5.4, it is clear that the above linear equation is embedded in the following circulant convolution

$$\hat{z}^c \otimes \hat{q}^c = \hat{r}^c \tag{5.186}$$

where $\hat{z}^c$ is a $2N$-periodic sequence obtained from the first row of the circulant matrices $\mathcal{Z}$; $\hat{r}^c$ and $\hat{q}^c$ are also $2N$-periodic sequences corresponding to the vectors r^c and q^c. Therefore, the linear equation can be solved by circulant deconvolution using the FFT with $O(NlogN)$ operations as

$$q^c = \mathcal{F}^{-1}\left[\frac{\tilde{\hat{r}}^c}{\tilde{\hat{z}}^c}\right] \tag{5.187}$$

where $\tilde{\hat{r}}^c, \tilde{\hat{z}}^c$ are the $2N$-point DFT of $\hat{r}^c$ and $\hat{z}^c$, respectively. The solution is

$$q = \text{the first } N \text{ elements of } q^c. \tag{5.188}$$

From the above derivation, we can find that the preconditioner is constructed in the spectral domain using the same spectral content of the system matrix A as we used in our CG-FFT algorithm. Thus no extra computer memory and computing time are required to derive this preconditioner. In addition, the preconditioning equation is solved efficiently in the spectral domain via a $2N$-point FFT. Therefore, we call the CG-FFT method combined with this preconditioning approach, *the Spectrally Preconditioned CG-FFT (SPCG-FFT) method* and $\tilde{\hat{z}}^c$ is called *the Spectral Preconditioner.*

To get the expression for the preconditioner in an $N \times N$ matrix format, we can re-organize equation 5.185. Since r and q are all $N \times 1$ vectors, there must be $N \times N$ matrices $\Delta^{(1)}$ and $\Delta^{(2)}$ existing, which make

$$\mho = \Delta^{(1)}r \qquad \wp = \Delta^{(2)}q, \tag{5.189}$$

where $\Delta^{(1)}, \Delta^{(2)}$ are diagonal matrices. The ith diagonal element of $\Delta^{(1)}$ is

$$\delta_i^{(1)} = \frac{\mho(i)}{r(i)}. \tag{5.190}$$

The elements of $\Delta^{(2)}$ can be solved for in a similar manner.

Putting these expressions into equation 5.185, we get

$$\left(A_N + A_N^e\Delta^{(2)}\right) q \;=\; r, \quad and \tag{5.191}$$

$$\left(A_N^e + A_N\Delta^{(2)}\right) q \;=\; \Delta^{(1)}r. \tag{5.192}$$

Compared with the formulation of PCG in section 5.2, we know that the preconditioner is

$$\mathcal{Z}_p^{(1)} = \; A_N + A_N^e\Delta^{(2)}, \quad or \tag{5.193}$$

$$\mathcal{Z}_p^{(2)} = \; (\Delta^{(1)})^{-1}(A_N^e + A_N)\Delta^{(2)} \tag{5.194}$$

Therefore, a large class of preconditioners can be constructed using this approach by making a selection between $\mathcal{Z}_p^{(1)}$ and $\mathcal{Z}_p^{(2)}$ along with using different values for $\Delta^{(i)}, i = 1, 2$. Here we study only those preconditioners that are of interest to us. In our study, the preconditioner $\mathcal{Z}_p^{(1)}$ is exploited because of its simplicity. The study of preconditioner $\mathcal{Z}_p^{(2)}$ can be carried by following the same procedure.

PRECONDITIONER #1 $\Delta^{(1)} = \Delta^{(2)} = \mathbf{I}$, $\mathbf{I}$ is an $N \times N$ Identity matrix. The preconditioner becomes

$$\mathcal{Z}_p^{(I)} = A_N + A_N^e$$

$$= \begin{pmatrix} a_0 & a_1 + a_{N-1} & a_2 + a_{N-2} & \dots & a_{N-2} + a_2 & a_{N-1} + a_1 \\ a_1 + a_{N-1} & a_0 & a_1 + a_{N-1} & \dots & a_{N-3} + a_3 & a_{N-2} + a_2 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{N-1} + a_1 & a_{N-2} + a_2 & a_{N-3} + a_3 & \dots & a_1 + a_{N-1} & a_0 \end{pmatrix}. \tag{5.195}$$

Surprisingly, $\mathcal{Z}_p^{(I)}$ is an $N \times N$ circulant matrix! This implies that if the preconditioner $\mathcal{Z}_p^{(I)}$ is used, we only need to perform an N-point circulant deconvolution to obtain the vector q

$$q = \mathcal{F}^{-1}\left[\frac{\tilde{r}}{\tilde{z}_1}\right], \tag{5.196}$$

where z_1 is the first row of $\mathcal{Z}_p^{(I)}$ and $\tilde{z}_1, \tilde{r}$ are the N-point DFTs of z_1 and r, respectively. This practice implies a great saving of computer memory and CPU time.

PRECONDITIONER #2 $\Delta^{(1)} = \Delta^{(2)} = -\mathbf{I}$, $\mathbf{I}$ is an $N \times N$ Identity matrix. The preconditioner becomes

$$\mathcal{Z}_p^{(-I)} = A_N - A_N^c$$

$$= \begin{pmatrix} a_0 & a_1 - a_{N-1} & a_2 - a_{N-2} & \dots & a_{N-2} - a_2 & a_{N-1} - a_1 \\ a_1 - a_{N-1} & a_0 & a_1 - a_{N-1} & \dots & a_{N-3} - a_3 & a_{N-2} - a_2 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{N-1} - a_1 & a_{N-2} - a_2 & a_{N-3} - a_3 & \dots & a_1 - a_{N-1} & a_0 \end{pmatrix}, \tag{5.197}$$

where $\mathcal{Z}_p^{(-I)}$ is a symmetric *Toeplitz* matrix. The first row of it is

$$\begin{cases} \{a_0, a_1 - a_{N-1}, a_2 - a_{N-2}, ..., a_\eta - a_{N-\eta}, -(a_\eta - a_{N-\eta}), ..., -(a_1 - a_{N-1})\} \\ \qquad \text{for odd } N: \quad \eta = \frac{1}{2}(N-1) \\ \{a_0, a_1 - a_{N-1}, a_2 - a_{N-2}, ..., a_\eta - a_{N-\eta}, 0, -(a_\eta - a_{N-\eta}), ..., -(a_1 - a_{N-1})\} \\ \qquad \text{for even } N: \quad \eta = \frac{1}{2}N - 1. \end{cases} \tag{5.198}$$

This type of *Toeplitz* matrix is called *skew circulant*. A skew circulant matrix can be converted into a circulant matrix by a transformation, i.e.,

$$V^H \mathcal{Z}_p^{(-I)} V \Rightarrow N \times N \text{ circulant matrix} \tag{5.199}$$

where V is a diagonal matrix. The details for this transformation can be found in [86]. Therefore, for the preconditioner $\mathcal{Z}_p^{(-I)}$, the preconditioning operation can also be performed using an N-point FFT.

PRECONDITIONER #3 Provided

$$\delta_i^{(1)} = \frac{r(N-i)}{r(i)}; \qquad \delta_i^{(2)} = \frac{q(N-i)}{q(i)}, \tag{5.200}$$

the vectors q and r are expanded in a wrap-around way into

$$q^e = \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ \cdot \\ \cdot \\ q_{N-1} \\ q_{N-1} \\ q_{N-2} \\ \cdot \\ \cdot \\ q_0 \end{pmatrix} \qquad \textit{and} \qquad r^e = \begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ \cdot \\ \cdot \\ r_{N-1} \\ r_{N-1} \\ r_{N-2} \\ \cdot \\ \cdot \\ r_0 \end{pmatrix}. \tag{5.201}$$

The preconditioner for this case, i.e., $\mathcal{Z}_p^{(++)}$, can be obtained by substituting $\delta_i^{(2)}$ into 5.193. It is not a *Toeplitz* circulant matrix. It is not even a symmetric matrix. The N-point FFT scheme used in PRECONDITIONERs #1 and #2 is not applicable to this case. The general $2N$-point FFT approach (5.181 - 5.188) has to be used here. Even though this preconditioner is not efficient compared with PRECONDITIONERs #1 and #2, for some cases, it may have better convergence performance compared with the above two preconditioners, $\mathcal{Z}_p^{(I)}$ and $\mathcal{Z}_p^{(-I)}$.

PRECONDITIONER #4 Correspondingly, another preconditioner can be obtained by choosing

$$\delta^{(1)} = -\frac{r(N-i)}{r(i)}; \qquad \delta^{(2)} = -\frac{q(N-i)}{q(i)}. \tag{5.202}$$

The vectors q and r are expanded using a negative wrap-around into

$$q^e = \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ \cdot \\ \cdot \\ q_{N-1} \\ -q_{N-1} \\ -q_{N-2} \\ \cdot \\ \cdot \\ -q_0 \end{pmatrix}; \qquad r^e = \begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ \cdot \\ \cdot \\ r_{N-1} \\ -r_{N-1} \\ -r_{N-2} \\ \cdot \\ \cdot \\ -r_0 \end{pmatrix}. \tag{5.203}$$

The $N \times N$ preconditioning matrix, $\mathcal{Z}_p^{(+-)}$, for this case is not *Toeplitz* circulant. It is not even a symmetric matrix. A $2N$-point FFT is used to carry out the preconditioning operation. However, the convergence performance of this preconditioner, for some cases, may be better than that of $\mathcal{Z}_p^{(I)}$ and $\mathcal{Z}_p^{(-I)}$.

5.5.3.2 Two-Dimensional Linear Structures - Block *Toeplitz* Preconditioners

When the problem to be analyzed consists of a structure where the flow of the current is only in one direction, the system matrix turns out to be a *BTT-matrix.* For instance, the microstrip array shown in Fig. 5.62 has N elements and the current is discretized into μ segments on each element. The resulting impedance matrix is

$$A = \begin{pmatrix} A_0 & A_1 & \dots & A_{N-1} \\ A_1 & A_2 & \dots & A_{N-2} \\ \dots & \dots & \dots & \dots \\ A_{N-1} & A_{N-2} & \dots & A_0 \end{pmatrix} \quad \text{where} \quad A_\kappa = \begin{pmatrix} a_{\kappa 0} & a_{\kappa 1} & \dots & a_{\kappa(\mu-1)} \\ a_{\kappa 1} & a_{\kappa 0} & \dots & a_{\kappa(\mu-2)} \\ \dots & \dots & \dots & \dots \\ a_{\kappa(\mu-1)} & a_{\kappa(\mu-2)} & \dots & a_{\kappa 0} \end{pmatrix} \tag{5.204}$$

with $\kappa = 0, ..., N-1$.

The preconditioning

$$Pq = r \Longrightarrow Aq = r \tag{5.205}$$

can be achieved by using two-dimensional circular deconvolution via a 2D FFT.

If we define the first row of A as

$$[\bar{a}_0, \bar{a}_1, ..., \bar{a}_{N-1}]$$

with $\bar{a}_\kappa = (a_{\kappa 0}, a_{\kappa 1}, ..., a_{\kappa(\mu-1)})$, $\kappa = 0, 1, ..., N-1$ denoting the first row of A_κ, and we define the vectors q and r as

$$[\bar{q}_1, \bar{q}_2, ..., \bar{q}_{N-1}]^T, \qquad [\bar{r}_1, \bar{r}_2, ..., \bar{r}_{N-1}]^T$$

with

$$\bar{q}_\kappa = [q_{\kappa 0}, q_{\kappa 1}, ..., q_{\kappa(\mu-1)}]^T \qquad \bar{r}_\kappa = [r_{\kappa 0}, r_{\kappa 1}, ..., r_{\kappa(\mu-1)}]^T,$$

we can construct three $(N-1) \times \mu$ matrices as

$$\mathcal{A} = \begin{pmatrix} \bar{a}_1 \\ \bar{a}_2 \\ . \\ . \\ \bar{a}_{N-1} \end{pmatrix}, \tag{5.206}$$

$$\mathcal{Q} = \begin{pmatrix} \bar{q}_1 \\ \bar{q}_2 \\ . \\ . \\ \bar{q}_{N-1} \end{pmatrix}, \quad and \tag{5.207}$$

$$\mathcal{R} = \begin{pmatrix} \bar{r}_1 \\ \bar{r}_2 \\ . \\ . \\ \bar{r}_{N-1} \end{pmatrix}. \tag{5.208}$$

In order to use the convolution theorem, we need to expand the above matrices. We increase their sizes by a factor of two in each dimension, following a wrap-around scheme. The expanded matrices are defined as

$$\mathcal{Z}^e(m,n) = \begin{cases} a_{m,n} & 0 \le m \le N-1, \ 0 \le n \le \mu - 1 \\ a_{m,2\mu-n} & 0 \le m \le N-1, \ \mu < n \le 2\mu - 1 \\ a_{2N-m,n} & N < m \le 2N-1, \ 0 \le n \le \mu - 1 \\ a_{2N-m,2\mu-n} & N < m \le 2N-1, \mu < n \le 2\mu - 1 \\ 0 & m = N \quad or \quad n = \mu \end{cases} \tag{5.209}$$

$$\mathcal{R}^e(m,n) = \begin{cases} r_{m,n} & 0 \le m \le N-1, \ 0 \le n \le \mu - 1 \\ \mathcal{R}_1(m,n) & N-1 < m \le 2N-1, 0 \le n \le \mu - 1 \\ \mathcal{R}_2(m,n) & 0 \le m \le N-1, \ \mu - 1 < n \le 2\mu - 1 \\ \mathcal{R}_3(m,n) & N-1 < m \le 2N-1, \mu - 1 < n \le 2\mu - 1 \end{cases} \tag{5.210}$$

$$\mathcal{Q}^e(m,n) = \begin{cases} q_{m,n} & 0 \le m \le N-1, \ \ 0 \le n \le \mu - 1 \\ \mathcal{Q}_1(m,n) & N-1 < m \le 2N-1, 0 \le n \le \mu - 1 \\ \mathcal{Q}_2(m,n) & 0 \le m \le N-1, \ \ \mu - 1 < n \le 2\mu - 1 \\ \mathcal{Q}_3(m,n) & N-1 < m \le 2N-1, \mu - 1 < n \le 2\mu - 1, \end{cases} \tag{5.211}$$

where $\mathcal{R}_\kappa, \mathcal{Q}_\kappa, \kappa = 1,2,3$ are arbitrary matrices. The preconditioning equation (5.205) can now be embedded in the following 2D circular convolution

$$\begin{aligned} \mathcal{R}^e(m,n) &= \mathcal{Z}^e(m,n) \otimes \mathcal{Q}^e(m,n) \\ &= \sum_{\alpha=-\infty}^{\infty} \sum_{\beta=-\infty}^{\infty} \mathcal{Z}^e\left[(m-\alpha)[\text{mod } 2N], (n-\beta)[\text{mod } 2\mu]\right] \\ &\cdot \ \mathcal{Q}^e\left[\alpha[\text{mod } 2N], \beta[\text{mod } 2\mu]\right]. \end{aligned} \tag{5.212}$$

Defining $\tilde{\mathcal{Z}}^e, \tilde{\mathcal{R}}^e$ and $\tilde{\mathcal{Q}}^e$ as the 2D DFT of $\mathcal{Z}^e, \mathcal{R}^e$ and $\mathcal{Q}^e$, respectively, we have

$$\tilde{\mathcal{Z}}^e \tilde{\mathcal{Q}}^e = \tilde{\mathcal{R}}^e \quad \Longrightarrow \quad \mathcal{Q}^e = \mathcal{F}^{-1}\left[\frac{\tilde{\mathcal{R}}^e}{\tilde{\mathcal{Z}}^e}\right]. \tag{5.213}$$

The solution of 5.205 is

$$q(\kappa) \approx \mathcal{Q}^e\left[\text{INT}\left(\frac{\kappa}{\mu-1}\right), m - \text{INT}\left(\frac{\kappa}{\mu-1}\right)\mu\right] \tag{5.214}$$

with $0 \le \kappa \le N\mu - 1$.

The expanded matrices $\mathcal{Z}^e, \mathcal{R}^e$ and $\mathcal{Q}^e$ are all $2N \times 2\mu$ matrices. Therefore two $2N \times 2\mu$ 2D FFTs and one $2N \times 2\mu$ 2D inverse FFT are necessary to generate the spectral preconditioner $\tilde{\mathcal{Z}}^e$ and carry out the preconditioning of the *BTT-matrix* in the spectral domain.

In order to obtain the matrix form of the preconditioner, we need to explore this problem from another direction.

It is not difficult, although tedious, to show that the 2D-circular convolution (5.212) is equivalent to the following matrix-vector multiplication

$$\begin{pmatrix} \aleph & \aleph_0 \\ \aleph_0 & \aleph \end{pmatrix} \begin{pmatrix} Q \\ Q_0 \end{pmatrix} = \begin{pmatrix} R \\ R_0 \end{pmatrix} \tag{5.215}$$

where $\aleph, \aleph_0$ are block-*Toeplitz* matrices. They are represented by their first block-row as

$$\aleph = Toeplitz\{A_0^e, A_1^e, ..., A_{N-1}^e\} \tag{5.216}$$

$$\aleph_0 = Toeplitz\{\emptyset, A_{N-1}^e, A_{N-2}^e, ..., A_1^e\} \tag{5.217}$$

with $\emptyset$ denoting a $2\mu \times 2\mu$ zero-matrix and

$$A_\kappa^e = \begin{pmatrix} A_\kappa & A_{\kappa 0} \\ A_{\kappa 0} & A_\kappa \end{pmatrix} \quad \kappa = 0, 1, ..., N-1 \tag{5.218}$$

$$A_{\kappa 0} = Toeplitz\{0, a_{\kappa(\mu-1)}, a_{\kappa(\mu-2)}, ..., a_{\kappa 1}\}. \tag{5.219}$$

The vectors Q, Q_0, R and R_0 are

$$Q_0 = \Theta Q; \qquad R_0 = \Upsilon R \tag{5.220}$$

$$Q = [q_0^e, q_1^e, q_2^e, ..., q_{N-1}^e]^T \tag{5.221}$$

$$R = [r_0^e, r_1^e, r_2^e, ..., r_{N-1}^e]^T \tag{5.222}$$

where

$$q_k^e = \begin{pmatrix} \bar{q}_\kappa \\ \bar{q}_{\kappa 0} \end{pmatrix}; \qquad r_k^e = \begin{pmatrix} \bar{r}_\kappa \\ \bar{r}_{\kappa 0} \end{pmatrix} \tag{5.223}$$

with

$$\bar{q}_{\kappa 0} = \alpha_\kappa \bar{q}_\kappa; \qquad \bar{r}_{\kappa 0} = \beta_\kappa \bar{r}_\kappa. \tag{5.224}$$

The quantities Θ, Υ and $\alpha_\kappa, \beta_\kappa$ are $2N\mu \times 2N\mu$ and $\mu \times \mu$ diagonal matrices, respectively.

From the above expression, it can be found that the expanded system matrix in 5.215 is a $2N \times 2N$ block circulant matrix with each block A^e_κ being a $2\mu \times 2\mu$ circulant matrix as well. Now we can carry out the same process as we did in Section 5.3.1.

In the first layer, equation (5.215) can be re-written as

$$(\aleph + \Theta\aleph_0)Q = R. \tag{5.225}$$

Defining $\Sigma = \aleph + \Theta\aleph_0$, we have

$$\Sigma = \begin{pmatrix} A^e_0 & A^e_1 + \theta_1 A^e_{N-1} & \dots & A^e_{N-1} + \theta_{N-1} A^e_1 \\ A^e_1 + \theta_0 A^e_{N-1} & A^e_0 & \dots & A^e_{N-2} + \theta_{N-1} A^e_2 \\ \dots & \dots & \dots & \dots \\ A^e_{N-1} + \theta_0 A^e_1 & A^e_{N-2} + \theta_1 A^e_2 & \dots & A^e_0 \end{pmatrix}, \tag{5.226}$$

where θ_κ, $\kappa = 0, 1, ..., N-1$ are the diagonal matrices decomposed from the diagonal matrix Θ.

In the second layer, we know that

$$\begin{aligned} & A^e_\kappa q^e_\kappa \Longrightarrow \\ & A^s_\kappa = A_\kappa + \alpha_\kappa A_{\kappa 0} \\ & = \begin{pmatrix} a_{\kappa 0} & a_{\kappa 1} + \alpha_{\kappa 1} a_{\kappa(\mu-1)} & \dots & a_{\kappa(\mu-1)} + \alpha_{\kappa(\mu-1)} a_{\kappa 1} \\ a_{\kappa 1} + \alpha_{\kappa 0} a_{\kappa(\mu-1)} & a_{\kappa 0} & \dots & a_{\kappa(\mu-2)} + \alpha_{\kappa(\mu-2)} a_{\kappa 2} \\ \dots & \dots & \dots & \dots \\ a_{\kappa(\mu-1)} + \alpha_{\kappa 0} a_{\kappa 1} & a_{\kappa(\mu-2)} + \alpha_{\kappa 1} a_{\kappa 2} & \dots & a_{\kappa 0} \end{pmatrix} \end{aligned} \tag{5.227}$$

Therefore Σ can be further reduced to

$$\mathcal{Z}^s = \begin{pmatrix} A_0^s & A_1^s + \theta_1 A_{N-1}^s & \dots & A_{N-1}^s + \theta_{N-1} A_1^s \\ A_1^s + \theta_0 A_{N-1}^s & A_0^s & \dots & A_{N-2}^s + \theta_{N-1} A_2^s \\ \dots & \dots & \dots & \dots \\ A_{N-1}^s + \theta_0 A_1^s & A_{N-2}^s + \theta_1 A_2^s & \dots & A_0^s \end{pmatrix}. \tag{5.228}$$

The preconditioning equation becomes

$$\mathcal{Z}^s q = r \tag{5.229}$$

and $\mathcal{Z}^s$ is, of course, the preconditioner in matrix form. It is an $N \times N$ block matrix and each block is a $\mu \times \mu$ matrix. A large class of preconditioners for *BTT-matrices* can be constructed using the above procedure by choosing different values for Θ, Υ, α and β. Generally, the matrix $\mathcal{Z}^s$ has no special structural advantages that we can exploit. The spectral preconditioning procedure described in (5.209 - 5.211) needs to be used to carry out the construction of spectral preconditioners and to perform the preconditioning efficiently in the spectral domain via the FFT. However, when some special values of Θ, Υ, α and β are chosen, the preconditioner $\mathcal{Z}^s$ can possess some special features which can help us to further improve the efficiency of this technique. Here we only give one example.

We choose Θ, Υ, α and β all to be Identity matrices. It is easy to show that A_κ^s and $\mathcal{Z}^s$ are circulant and block circulant, respectively. That is, the matrix form preconditioner $\mathcal{Z}^s$ consists of an $N \times N$ block circulant matrix, and each block is a $\mu \times \mu$ circulant matrix. Therefore, to solve this set of preconditioning equations, the expansion process is not necessary. Two-dimensional FFT processing can be directly performed on both side of this equation. Specifically, if σ is an $N \times \mu$ matrix composed of the first-row elements of $\mathcal{Z}^s$, then

$$\tilde{\sigma}\tilde{\mathcal{Q}} = \tilde{\mathcal{R}} \quad \Longrightarrow \quad \mathcal{Q} = \mathcal{F}^{-1}\left[\frac{\tilde{\mathcal{R}}}{\tilde{\sigma}}\right]. \tag{5.230}$$

The quantities, $\tilde{\sigma}, \tilde{\mathcal{R}}$ and $\tilde{\mathcal{Q}}$ are the 2D DFTs of $\sigma, \mathcal{R}$ and $\mathcal{Q}$, respectively. The solution of the preconditioning equation is

$$q(\kappa) \approx \mathcal{Q}\left[\mathrm{INT}\left(\frac{\kappa}{\mu-1}\right), m - \mathrm{INT}\left(\frac{\kappa}{\mu-1}\right)\mu\right] \tag{5.231}$$

with $0 \leq \kappa \leq N\mu - 1$.

Compared with the general preconditioning procedure where $2N \times 2\mu$ 2D DFTs are employed, the present treatment only needs $n \times \mu$ 2D DFTs. A considerable saving of computer memory and computing time can be achieved using this preconditioner.

5.5.3.3 Two-Dimensional Planar Structures - General Preconditioners

Up to now, all the preconditioners we designed are for the EM analysis of linear 1D or 2D structures, that is, the current on the structures flows in only one direction (such as the approximate formulations that are used for single narrow microstrip lines, or microstrip dipole arrays). Now, we are going to design the preconditioners for planar structures with the current flowing in variable directions. Eventually, the current can be decomposed into two components (x- and y-directed components) with mutual coupling between them. The simplest example of this case is that of scattering from a rectangular microstrip patch.

We have shown that the system matrix used for the analysis of planar structures is a 2×2 block *BTT-matrix* as shown in (5.172). The preconditioning equation for this case can be written as

$$\begin{pmatrix} A_{xx} & A_{xy} \\ A_{xy} & A_{yy} \end{pmatrix} \begin{pmatrix} q_x \\ q_y \end{pmatrix} = \begin{pmatrix} r_x \\ r_y \end{pmatrix}. \tag{5.232}$$

Since A_{xx}, A_{xy} and A_{yy} are all *BTT-matrices*, we can treat each of them separately as we treated A in the last section. The result of this treatment is similar to that obtained above:

$$\mathcal{Z}^e_{xx} \otimes \mathcal{Q}^e_x + \mathcal{Z}^e_{xy} \otimes \mathcal{Q}^e_y = \mathcal{R}^e_x \tag{5.233}$$

$$\mathcal{Z}^e_{xy} \otimes \mathcal{Q}^e_x + \mathcal{Z}^e_{yy} \otimes \mathcal{Q}^e_y = \mathcal{R}^e_y. \tag{5.234}$$

The construction of $\mathcal{Z}^e_{xx}, \mathcal{Z}^e_{xy}$ and $\mathcal{Z}^e_{yy}$ follows the same procedure that was used for deriving (5.209), using A_{xx}, A_{xy} and A_{yy}, respectively. The quantities $\mathcal{Q}^e_x, \mathcal{Q}^e_y, \mathcal{R}^e_x$ and $\mathcal{R}^e_y$ are $2M \times 2N$ matrices. They are constructed using q_x, q_y, r_x and r_y by following the same procedure that was used for deriving $\mathcal{R}^e$ and $\mathcal{Q}^e$ in 5.210 and 5.211.

In the spectral domain, (5.234) can be written as

$$\tilde{\mathcal{Z}}^e_{xx} \cdot \tilde{\mathcal{Q}}^e_x + \tilde{\mathcal{Z}}^e_{xy} \cdot \tilde{\mathcal{Q}}^e_y = \tilde{\mathcal{R}}^e_x, \tag{5.235}$$

$$\tilde{\mathcal{Z}}^e_{xy} \cdot \tilde{\mathcal{Q}}^e_x + \tilde{\mathcal{Z}}^e_{yy} \cdot \tilde{\mathcal{Q}}^e_y = \tilde{\mathcal{R}}^e_y. \tag{5.236}$$

Thus, the solution of this spectral-domain linear equation is easily obtained as

$$\tilde{\mathcal{Q}}^e_x = \frac{\tilde{\mathcal{Z}}^e_{yy} \cdot \tilde{\mathcal{R}}^e_x - \tilde{\mathcal{Z}}^e_{xy} \cdot \tilde{\mathcal{R}}^e_y}{\tilde{\mathcal{Z}}^e_{yy} \cdot \tilde{\mathcal{Z}}^e_{xx} - \tilde{\mathcal{Z}}^e_{xy} \cdot \tilde{\mathcal{Z}}^e_{xy}}, \quad \textit{and} \tag{5.237}$$

$$\tilde{\mathcal{Q}}^e_y = \frac{\tilde{\mathcal{Z}}^e_{xx} \cdot \tilde{\mathcal{R}}^e_y - \tilde{\mathcal{Z}}^e_{xy} \cdot \tilde{\mathcal{R}}^e_x}{\tilde{\mathcal{Z}}^e_{yy} \cdot \tilde{\mathcal{Z}}^e_{xx} - \tilde{\mathcal{Z}}^e_{xy} \cdot \tilde{\mathcal{Z}}^e_{xy}}. \tag{5.238}$$

The solution of the preconditioning equation 5.232 can be obtained by simply carrying out the 2D inverse Fourier transforms of $\tilde{\mathcal{Q}}^e_x$ and $\tilde{\mathcal{Q}}^e_y$ via the FFT, truncating the results and then re-organizing Q^e_x and Q^e_y to get the solution q_x and q_y.

Now we can say that the spectral domain preconditioning can be performed to simulate 2D planar structures. In this instance, there are three spectral preconditioners which are $\tilde{\mathcal{Z}}^e_{xx}, \tilde{\mathcal{Z}}^e_{xy}$ and $\tilde{\mathcal{Z}}^e_{yy}$. The 2D DFTs involved in this case consist of four $2M \times 2N$ 2D DFTs and two $2M \times 2N$ 2D inverse DFTs.

The matrix form preconditioner can also be derived from the spectral preconditioner. It has the following form:

$$\mathcal{Z}^s = \begin{pmatrix} \mathcal{Z}^s_{xx} & \mathcal{Z}^s_{xy} \\ \mathcal{Z}^s_{xy} & \mathcal{Z}^s_{yy} \end{pmatrix}, \tag{5.239}$$

where $\mathcal{Z}^s_{xx}, \mathcal{Z}^s_{xy}$ and $\mathcal{Z}^s_{yy}$ are the matrix preconditioners corresponding to the spectral preconditioners $\tilde{\mathcal{Z}}^e_{xx}, \tilde{\mathcal{Z}}^e_{xy}$ and $\tilde{\mathcal{Z}}^e_{yy}$, respectively. Generally, this is not a symmetric matrix. The matrix-extending process has to be adopted here in order to employ the circular convolution/deconvolution. But, when we choose preconditioners as we did in 5.230, we can achieve a considerable saving in computer memory and computing time. The DFTs involved for this preconditioner consist of four $M \times N$ 2D DFTs and two $M \times N$ 2D inverse DFTs in contrast to four $2M \times 2N$ 2D DFTs and two $2M \times 2N$ 2D inverse DFTs for the general preconditioning cases.

5.6 Summary

In this chapter, a new highly efficient spectrally preconditioned CG-FFT method has been presented. Starting with the Strang and Chan *Toeplitz* preconditioners, a new systematic approach for designing a new class of spectral *Toeplitz* preconditioners is developed. Further, this approach has been extended from simple one-dimensional problems to two-dimensional linear problems and then two-dimensional planar problems. We described an approach for designing non-*Toeplitz* preconditioners. This method takes the full advantage of the FFT by carrying out the CG operation, constructing the preconditioner and operating the preconditioning all in the spectral domain. It has the advantage that no additional memory and only $O(NlogN)$ additional operations are needed for storing and constructing the preconditioner and performing the preconditioning.

The analysis of the spectrum, convergence performance and numerical experiments will be presented in the next chapter.

Chapter 6

Spectra of Spectral Preconditioners

We already know that a good preconditioner for A is a matrix P that approximates A well in the sense that the spectrum of the preconditioned matrix $P^{-1}A$ is clustered in a small region around $(1,0)$ in the complex plane. Thus the spectral properties, that is the distributions of the eigenvalues of the preconditioned matrices, can be taken as criterion with which to judge the performance of the preconditioners. In this section, the spectral properties of the preconditioners developed by Strang and Chan are compared with a preconditioner that has been developed during the cause of this study. Some numerical examples of the applications of these preconditioners for EM analysis are also presented in this chapter.

6.1 Analysis of performance of preconditioners for 1D structures

As a demonstration of the use of SPCG for 1D EM analysis, we analyze the characteristics of microstrip lines. Consider a narrow microstrip line with a width $w = 0.005\lambda$ and length $L = 2\lambda$ as shown in Fig. 5.62. Z-directed roof-top functions can be used to discretize the current on the strip. If the number of segments $N = 79$, the system matrix (i.e., impedance matrix) is a 79×79 *Toeplitz* matrix. The three preconditioners developed by Strang, Chan and ourselves are all applied to this system matrix. The spectra of the system matrix A and the preconditioned matrices $\mathcal{S}^{-1}A, \mathcal{C}^{-1}A, (\mathcal{Z}_p^{(l)})^{-1}A, (\mathcal{Z}_p^{(-l)})^{-1}A, (\mathcal{Z}_p^{(++)})^{-1}A$ and $(\mathcal{Z}_p^{(+-)})^{-1}A$ are calculated. In Fig. 6.63 are illustrated the spectra of $\mathcal{S}^{-1}A, \mathcal{C}^{-1}A$ and $(\mathcal{Z}_p^{(l)})^{-1}A$. It can be seen that the spectra of all the preconditioned matrices are clustered around $(1, 0)$ while the spectrum of the original matrix A is widely spread over a large region in the complex plane. The eigenvalues and the condition numbers of A and the preconditioned matrices are listed in Appendix B, Table B.1. These results indicate that for the *Toeplitz* system that arises from EM analysis of 1D structures, the preconditioners $\mathcal{S}, \mathcal{C}$ and $\mathcal{Z}_p$ can improve the spectral properties of the system matrix, but to varying degrees.

To further compare the properties of these different preconditioners, we plot a blown-up view of the spectra of $\mathcal{S}^{-1}A, \mathcal{C}^{-1}A$ and $(\mathcal{Z}_p^{(l)})^{-1}A$ in Fig. 6.64. From this graph, we conclude that the spectrum of Chan preconditioner is spread out beyond (1,0). Strang and the proposed preconditioner have similar spectra. Actually, the proposed one is a little bit better. They only have 4 outliers located near (2.6,-0.5), (0.4,0.05), (1.0,0.25) and (0.8,0.1). All the other 75 eigenvalues are closely clustered around (1,0). This result shows that the proposed and Strang

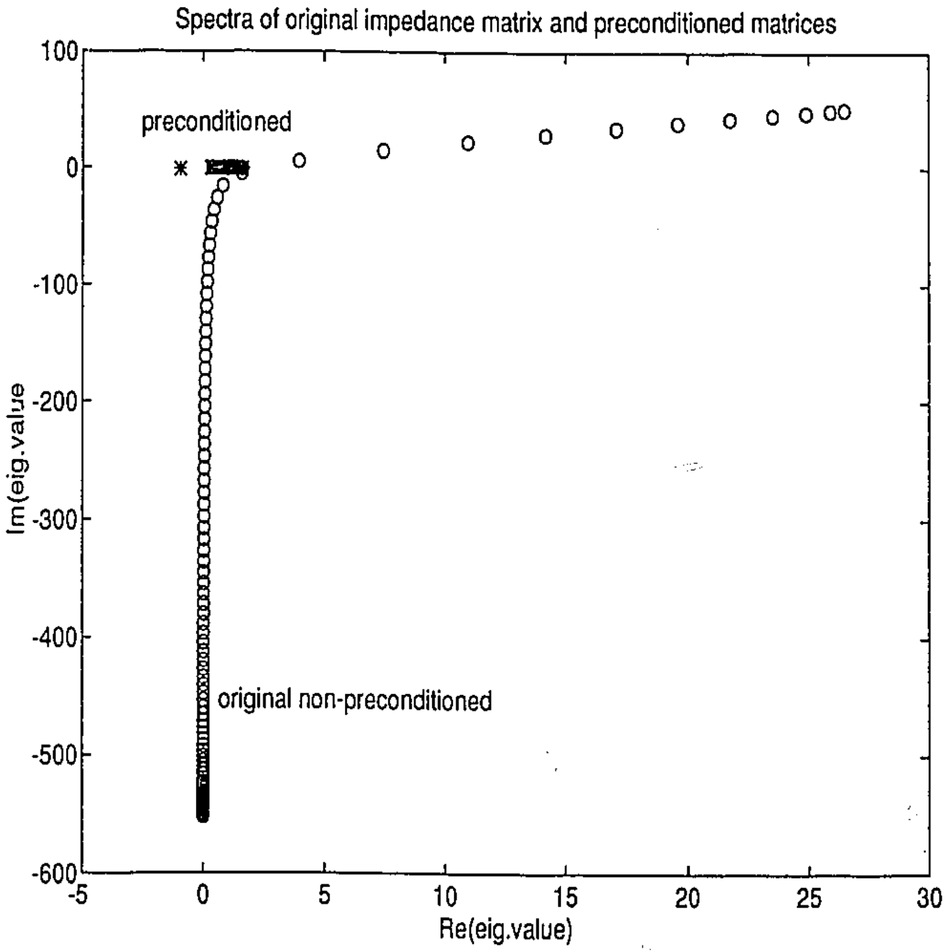


Figure 6.63: The spectra of system matrix A and the preconditioned matrices $\mathcal{S}^{-1}A, \mathcal{C}^{-1}A, (\mathcal{Z}_p^{(I)})^{-1}A$

preconditioners may have similar performance which is much better than the Chan preconditioner when they are used to treat system matrices for 1D structures. The convergence rates of the solutions using different preconditioners are illustrated in Fig. 6.71 which agrees with the results given above.

The spectra of the preconditioned matrices, based on four options for the proposed preconditioners, i.e., $(\mathcal{Z}_p^{(I)})^{-1}A, (\mathcal{Z}_p^{(-I)})^{-1}A, (\mathcal{Z}_p^{(++)})^{-1}A$ and $(\mathcal{Z}_p^{(+-)})^{-1}A$ are shown in Fig. 6.65. The condition numbers of these four preconditioned matrices are listed in Appendix B, Table B.2. It can be found for this example that among the four preconditioners, the $\mathcal{Z}_p^{(+-)}$ has the best performance. That is, it has the smallest condition number and most clustered spectrum. The convergence rates of the corresponding four solutions are shown in Fig. 6.72. They also support this assertion.

6.2 Analysis of performance of preconditioners for 2D Structures

As an example, an array of microstrip lines is analyzed to study the spectra of different preconditioners for BTT matrices. Three types of BTT preconditioners have been compared. They are Strang BTT preconditioner and the proposed BTT preconditioners #1 and #2.

The Strang preconditioner is generated using the same principle as was applied to the Strang *Toeplitz* preconditioner. If the system matrix is an $N \times N$ BTT matrix with each of the block a $\mu \times \mu$ *Toeplitz* matrix, the Strang BTT preconditioner approximates this matrix with an $N \times N$ block-circulant matrix which has $\mu \times \mu$ circulant blocks. Therefore the preconditioning operation can be

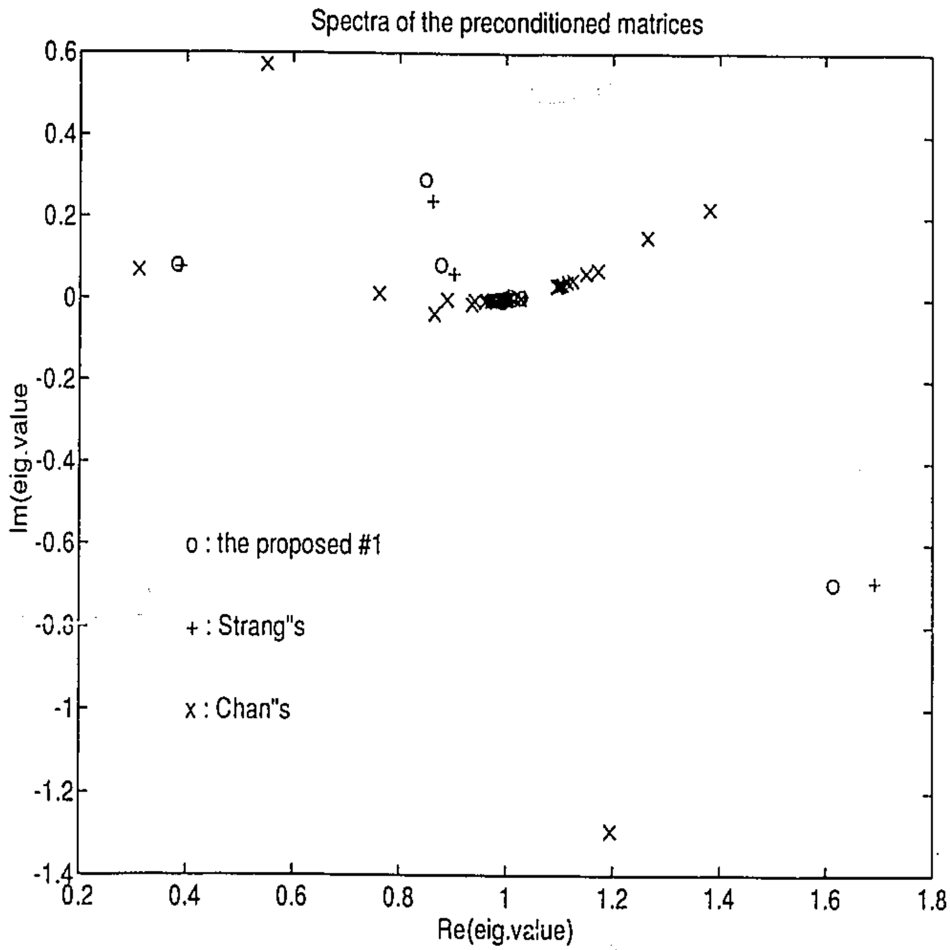


Figure 6.64: A blown-up view of the spectra of the preconditioned matrices $\mathcal{S}^{-1}A, \mathcal{C}^{-1}A$ and $(\mathcal{Z}_p^{(l)})^{-1}A$. "o" : proposed #4; "+","x" and "*" : others.

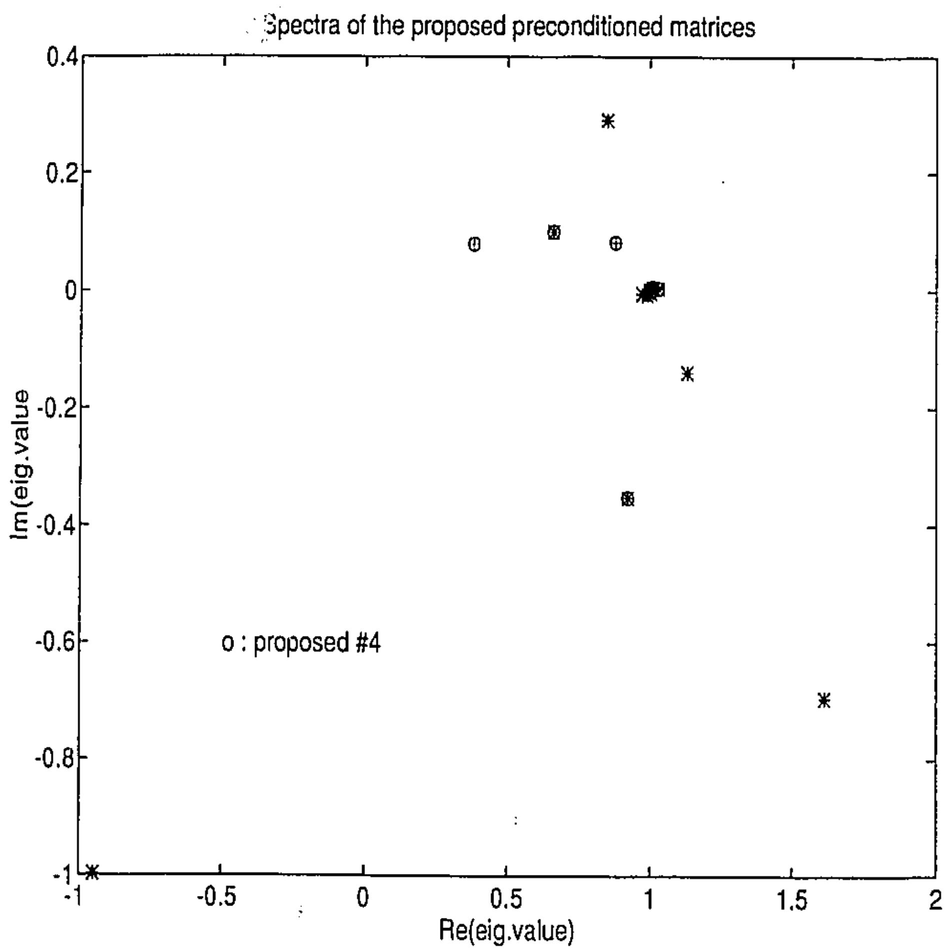


Figure 6.65: The spectra of the preconditioned matrices $(\mathcal{Z}_p^{(I)})^{-1}A, (\mathcal{Z}_p^{(-I)})^{-1}A$, $(\mathcal{Z}_p^{(++)})^{-1}A$ and $(\mathcal{Z}_p^{(+-)})^{-1}A$

easily performed using an $N \times \mu$ point 2D FFT.

Proposed preconditioner #1 is constructed by choosing Θ, Υ, α and β all to be Identity matrices. This preconditioner is also based on an $N \times N$ block circulant matrix with $\mu \times \mu$ circulant blocks. An $N \times \mu$ point 2D FFT is also used to carry out this preconditioning.

Proposed preconditioner #2 is constructed by choosing Θ, Υ, α and β all to be elementary matrices. This preconditioner is neither a circulant nor a block circulant matrix. A $2N \times 2\mu$ point 2D FFT must be used to carry out the preconditioning after doing the expansion given in 5.209 - 5.211.

The spectra of the original matrix and the preconditioned matrices are shown in Fig. 6.66. The proposed preconditioned matrices show a strong clustering feature compared to the original one. But the Strang preconditioner is less clustered, even though it improves the original spectrum. Fig. 6.67 shows a blown-up version of the previous figure with the original spectrum omitted. It is clear that the Strang preconditioner's spectrum is spread over a much larger region in the complex plane than is the case for the proposed preconditioners. This implies that the Strang preconditioner may not have good performance when it is applied to a 2D problem. The numerical examples given in the next section will also help to illustrate this point.

The Strang preconditioner works well for 1D analysis but not well for the 2D case. This is attributable to the principle used in the construction of the Strang preconditioner and the structural properties of the impedance matrices arising from the EM analysis. In EM analysis, the elements of the impedance matrices are inversely proportional to the distance between the testing and the basis functions raised to a power greater than or equal to one-half. For 1D structures, if the

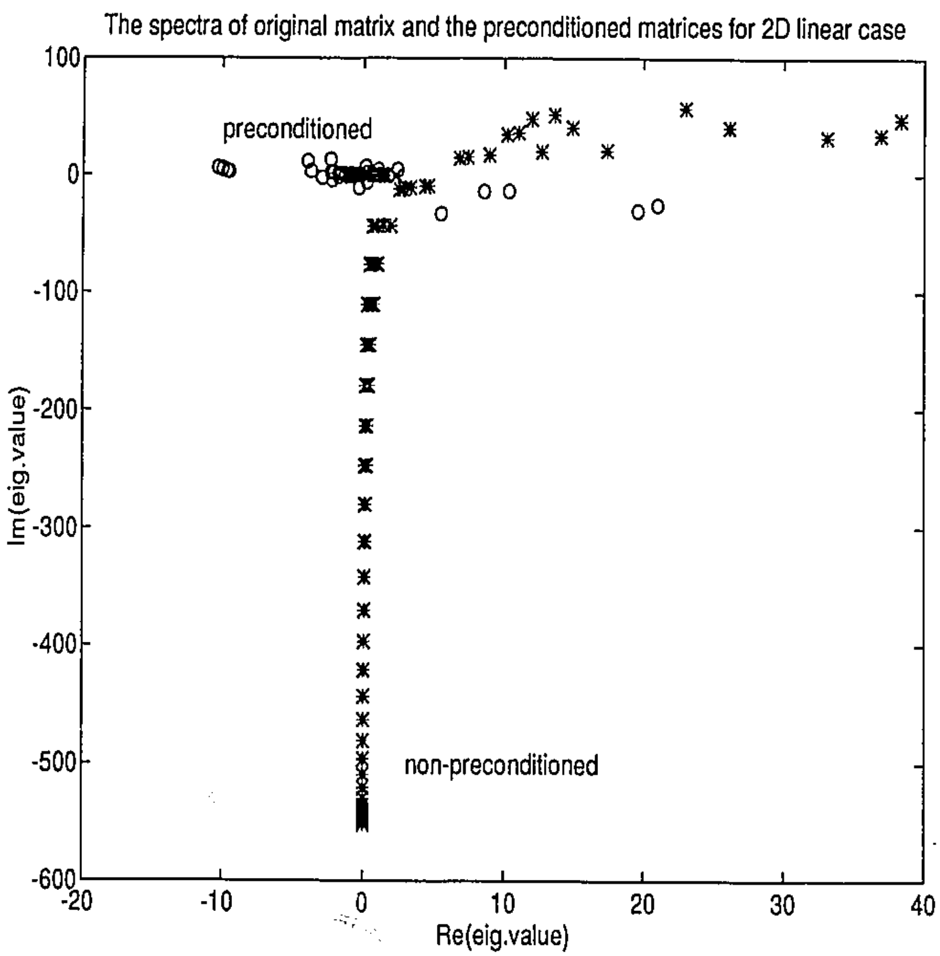


Figure 6.66: The spectra of original matrix and the preconditioned matrices for 2D linear structure analysis

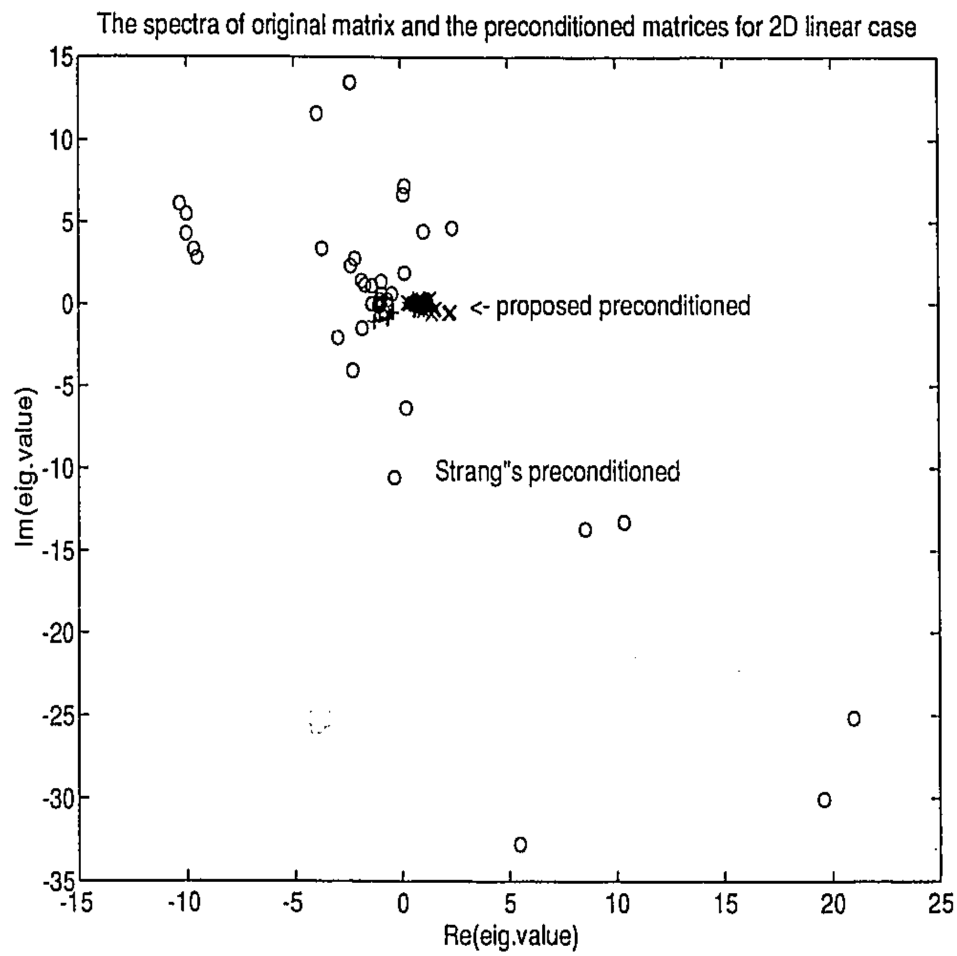


Figure 6.67: A blown-up view of Fig. 6.66 (the spectrum of original matrix is omitted.)

discretized segments are sequentially numbered, the magnitude of the elements of the matrix can be made to decay with the "distance" as one moves away from the diagonal. In the construction of a Strang preconditioner, a circulant matrix is developed by replicating the central diagonals from the original matrix to complete the circulant (see Section 5.5.1 for details). The error matrix has non-zero elements only in the upper-right and lower-left corners. Since the original matrix has diagonal magnitudes decaying as one moves away from the diagonal, the error matrix is minimized in the infinite norm [16]. When a 2D structure is analyzed, the impedance matrix is a BTT matrix. The elements of the matrix do not decay monotonically with distance from the diagonal. The off-diagonal blocks have relatively large elements, especially along the diagonal of the off-diagonal blocks. The poor performance of the Strang preconditioner for 2D analysis can be attributed to this fact. The proposed preconditioners are constructed in a totally different way. All the elements of the original matrix are used in the construction. They contain more information about the original matrix than the Strang preconditioner does. And they do not rely on the special structural properties of the original matrix, such as off-diagonal-decaying. Therefore the proposed preconditioners work equally well in 1D and 2D cases.

To compare the spectra of the different proposed preconditioners, a blown-up version of Fig. 6.67 is shown in Fig. 6.68 with the Strang spectrum omitted. We can see that they are clustered in a very small region ([-1.5:2.5],[-1:0.4]) compared with that of Strang ([-15:25],[-35:15]) and the original one ([0:40],[-600:100]). Among the four proposed preconditioners, #3 has the greatest clustering of its spectrum. It spreads over only a tiny region ([0.25:1.2],[-0.35:0.2]). We can expect that it also has the best rate of convergence.

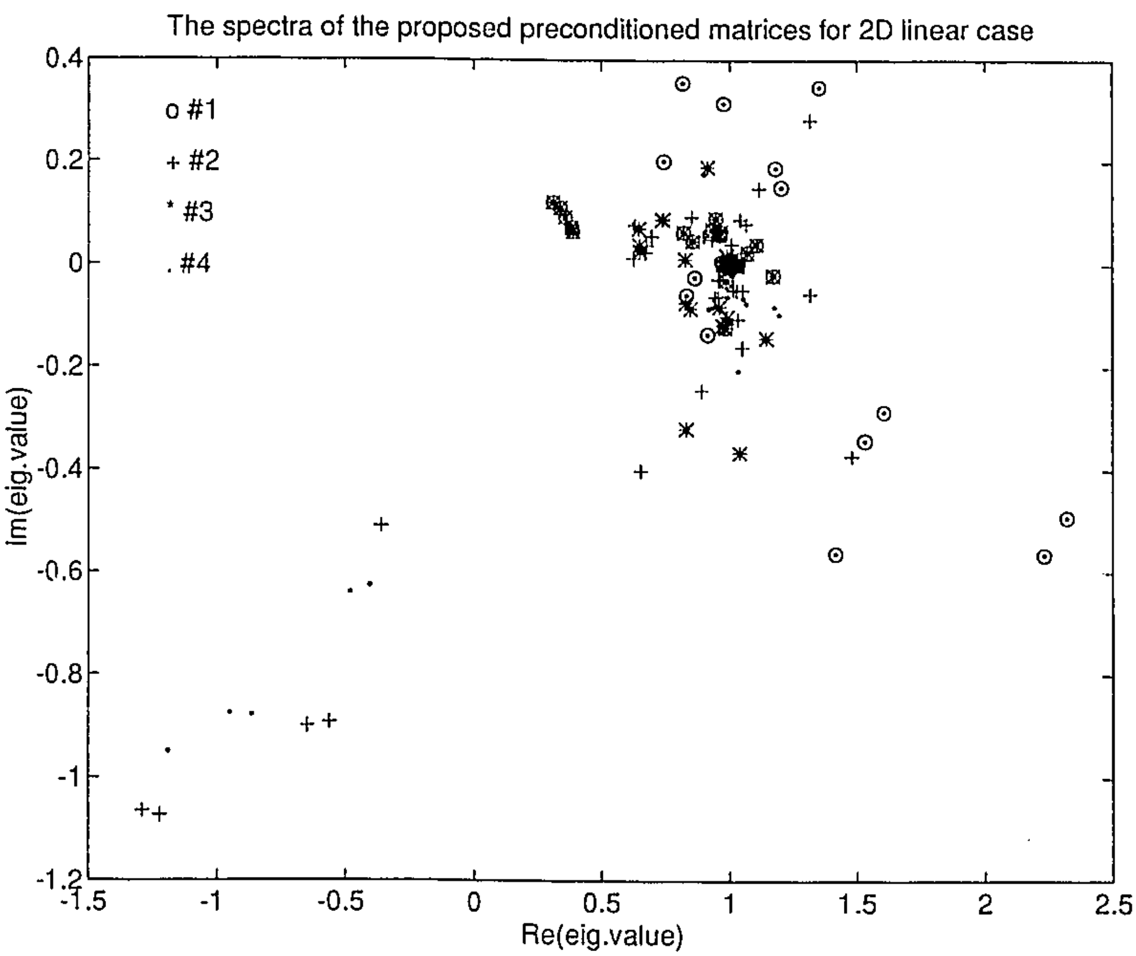


Figure 6.68: A further blown-up view of Fig.6.66 (Strang's spectrum is omitted.)

6.3 Numerical Results and Comparison of Convergence Performance

As a further demonstration of the SPCG-FFT method, several EM structures have been analyzed.

First, the current distribution on a long microstrip line is simulated using the Method of Moments, CG-FFT and SPCG-FFT. The configuration of the structure is illustrated in Fig.5.62. It is driven by a δ-gap source at one end of the microstrip line. In the first example, 199 narrow roof-top functions are used to expand the current. Figs.6.69 and 6.70 give the simulated results obtained by the Method of Moments, CG-FFT and SPCG-FFT. All of them are in excellent agreement. Fig.6.71 shows the convergence rates of the CG-FFT and different versions of the SPCG-FFT. The ERROR is defined as a 2-norm of the residual (i.e. $||r||_2$). It is obvious that the proposed SPCG-FFT has better performance than the others. The Strang preconditioner is close to the proposed one. The Chan preconditioner is worse than the other two even though it still improves the convergence rate of the non-preconditioned CG-FFT. For the non-preconditioned CG-FFT, a steep decline of the convergence rate occurs after 90 iterations (beyond the graph) while the proposed SPCG-FFT achieves an error of 10^{-20} just after 12 iterations. This is a remarkable difference. Fig.6.72 shows the convergence rates obtained by using different proposed preconditioners. Of the four preconditioners to be tested, #4 has the best performance. This is consistent with its spectral properties which were described in the last section, where it achieved the greatest degree of clustering from among these four preconditioners.

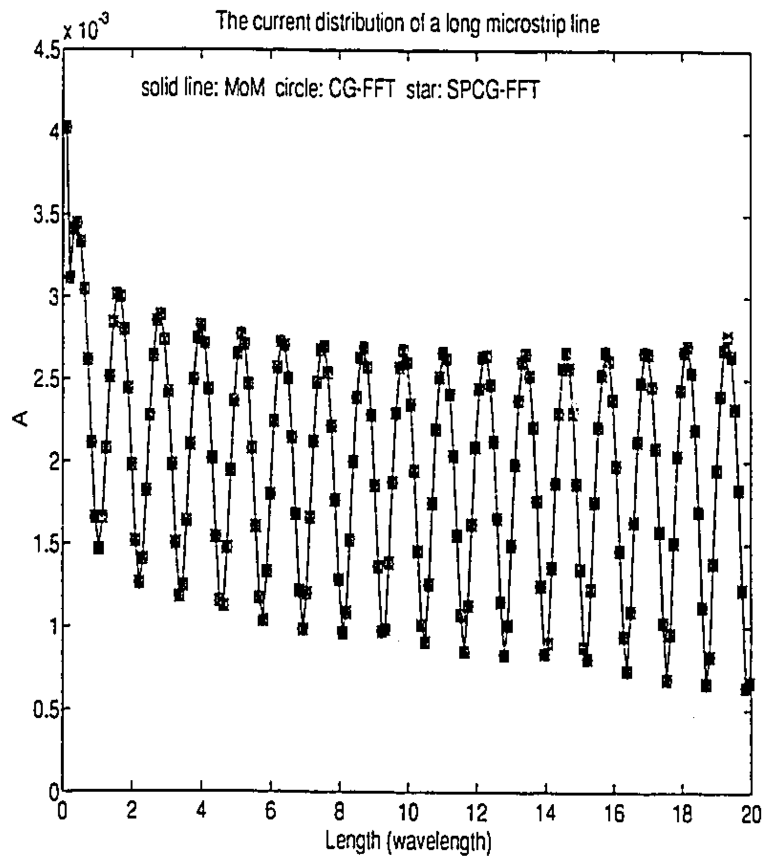


Figure 6.69: The magnitude of the current distribution on a long microstrip line

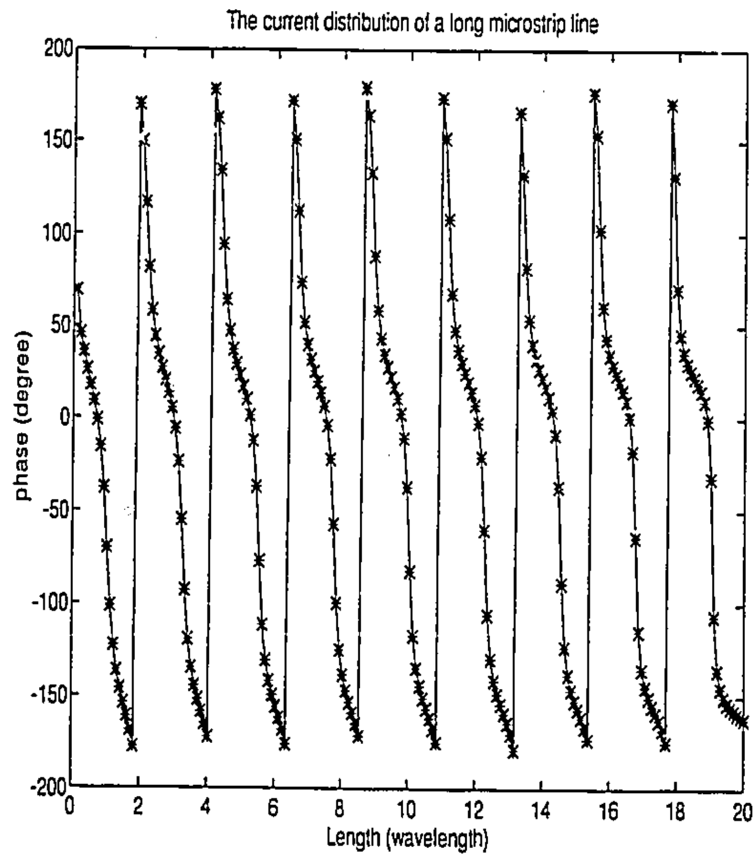
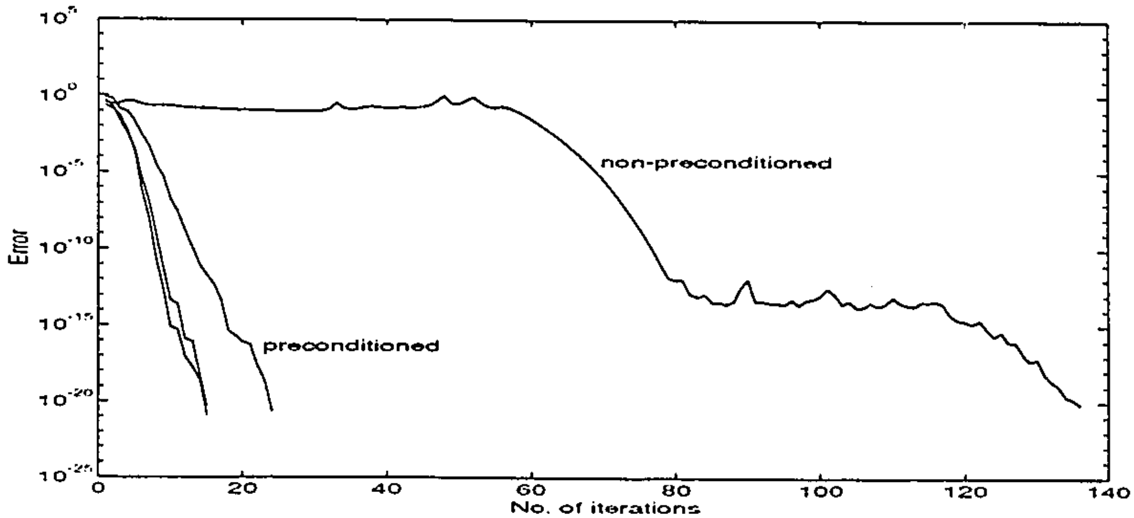


Figure 6.70: The phase of the current distribution on a long microstrip line



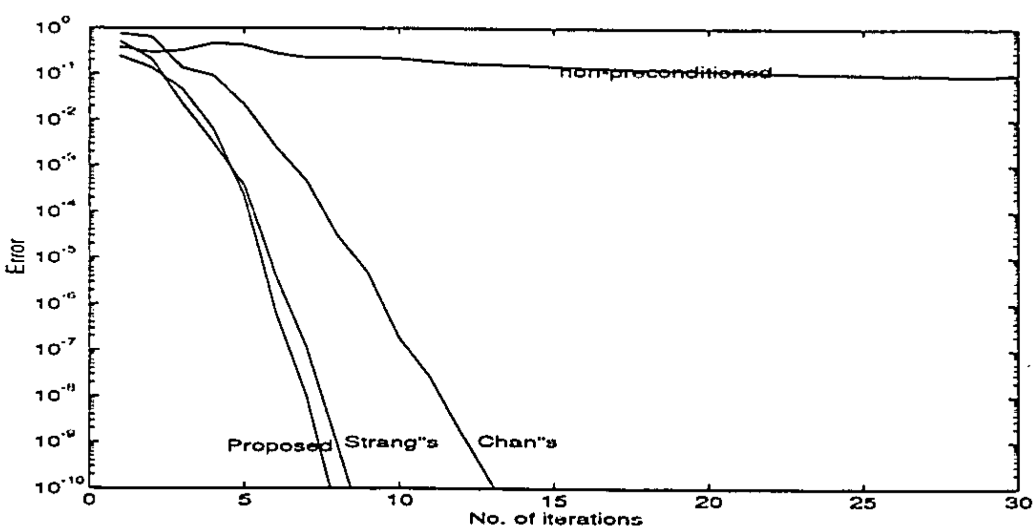


Figure 6.71: The convergence rates of the CG-FFT and different versions of the SPCG-FFT (lower one is the blown-up view of the upper one.)

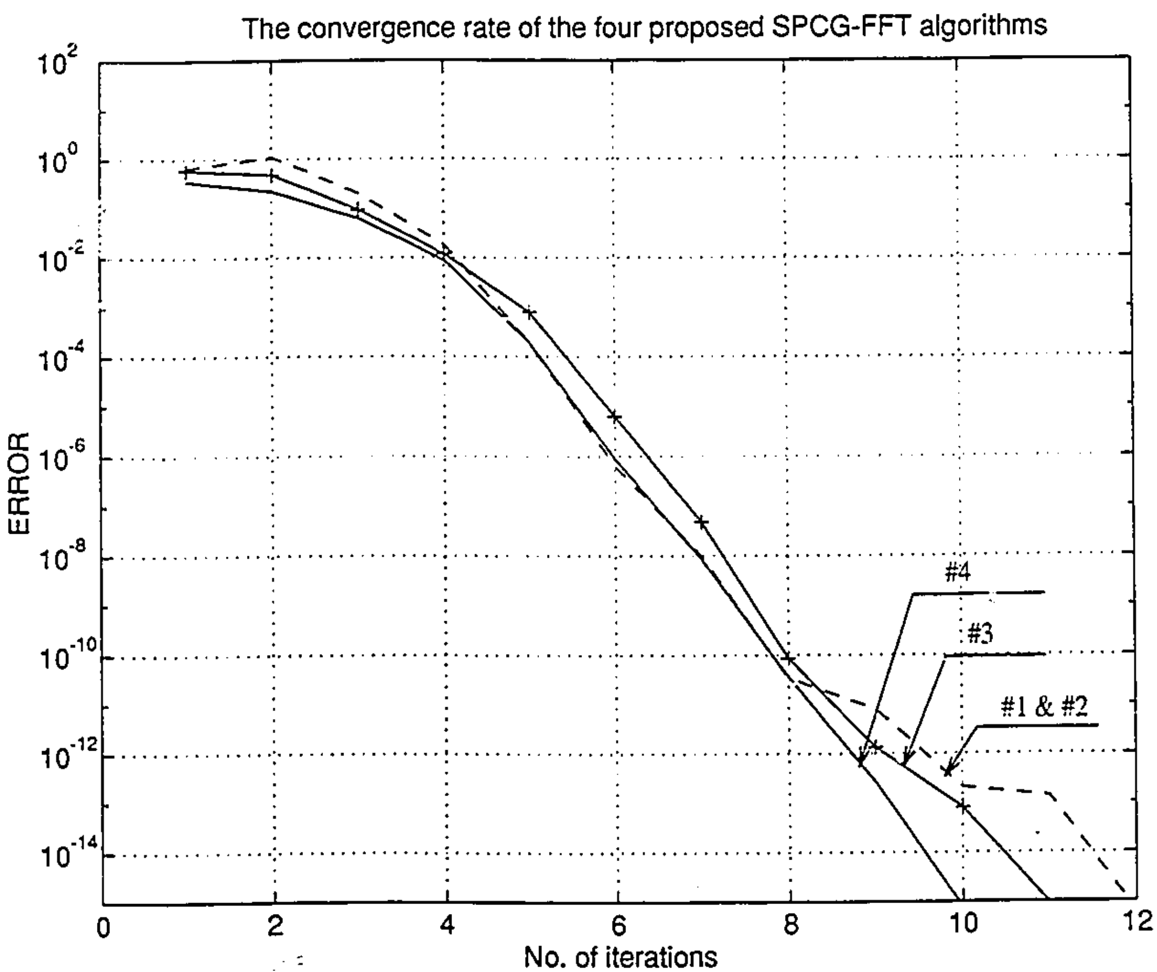


Figure 6.72: The rates of convergence of the four proposed SPCG-FFT algorithms

To further investigate the convergence behavior of the SPCG-FFT algorithm, the microstrip line in the first example is discretized and simulated using different numbers of basis functions. Table.6.2 lists the run times and iteration performance of the non-preconditioned CG-FFT method, the Strang PCG-FFT method and the proposed SPCG-FFT method.

From the CG-FFT algorithm, we know that the computation time per iteration T_i follows the relation

$$T_i = CN_s \log N_s \tag{6.240}$$

with N_s denoting the number of unknowns and C being a constant related to the number of FFTs performed during each iteration. The total computation time T required to solve the problem follows the relation

$$T = N_i T_i = CN_i N_s \log N_s, \tag{6.241}$$

where N_i denotes the number of iterations required to arrive at a solution. Theoretically, the CG algorithm fully converges with at most N_s iterations. Thus the computation time for a solution can be as large as

$$T = CN_s^2 \log N_s. \tag{6.242}$$

However, from the above numerical experiments, it can be found that the number of iterations N_i required to obtain a fixed error (e.g. $||r||_2 \leq 10^{-5}$, which is good enough for most engineering problems) is much smaller than the number of unknowns. In order to understand the asymptotic behavior of these algorithms, graphs of $\log_2(T)$ vs. $\log_2(N_s)$ are plotted in Fig.6.73. A measure of the asymptotic performance for large N_s for each algorithm can be determined by taking the gradients of the lines fitted to the last three points on each graph. We observe that the computation time for the non-preconditioned CG-FFT grows roughly as

Table 6.2: Iteration performance of different CG-FFT algorithms

The proposed algorithm

N_s	16	32	64	128	256	512	1024	2048	4096
N_i	5	6	6	6	7	7	7	7	7
$CPU(Sec.)$	0.1162	0.1500	0.2331	0.3500	0.6177	1.1333	2.2600	4.8160	10.9000

Strang's algorithm

N_s	16	32	64	128	256	512	1024	2048	4096
N_i	5	6	6	6	7	7	7	7	7
$CPU(Sec.)$	0.1667	0.2000	0.3501	0.3667	7.6177	1.4833	2.7634	4.8160	15.4500

Chan's algorithm

N_s	16	32	64	128	256	512	1024	2048	4096
N_i	7	9	9	9	9	9	9	9	9
$CPU(Sec.)$	0.3328	0.4368	0.4991	0.5889	0.9036	1.5998	3.9901	6.5488	15.5231

Non-preconditioned CG-FFT algorithm

N_s	16	32	64	128	256	512	1024	2048	4096
N_i	12	22	39	70	129	244	782	——	——
$CPU(Sec.)$	0.2833	0.4000	0.9000	2.7000	8.6833	28.250	100.38	——	——

$O(N_s^{\frac{3}{2}} \log N_s)$. On the other hand, in the case of the PCG-FFT methods, the computation times are $O(N_s \log N_s)$. Comparing these with relation 6.241, we see that the number of iterations for CG-FFT is roughly proportional to the square root of the number of unknowns, i.e., $N_i \approx O(\sqrt{N_s})$, while the computing times for PCG-FFT are independent of the number of unknowns. This observation can be confirmed by the almost constant number of iterations for PCG-FFT algorithms in Table.6.2.

In the next example, we carry out a simulation of a microstrip line array. The array is composed of 5 long microstrip lines with inter-element spacing of 0.6λ. Each line is 6.45λ long. Only the first microstrip line is driven by a δ-gap source which is located at the end of line. 128 narrow roof-top basis functions were used to expand the current on each line. Totally there are 640 unknowns. This problem was simulated using the Method of Moments, the CG-FFT, the Strang PCG-FFT and the proposed SPCG-FFT. In Fig. 6.74 is shown the magnitude of the current distribution on the first three microstrip lines. The graph with the circles gives the results obtained using the Method of Moments. The results obtained using the Strang and the proposed PCG-FFT totally overlap with each other. They are represented by the solid lines in the figure. All the results are in excellent agreement. Fig. 6.75 shows the curves of convergence rates for the four different CG algorithms. We observe that the non-preconditioned CG-FFT algorithm has the slowest rate of convergence. Strang's preconditioning algorithm can improve the performance of the CG-FFT but not as good as the proposed ones. The proposed preconditioning approach #3 has the best performance among all the preconditioners, especially in the case of more restrictive convergence requirements ($||r|| \leq 10^{-10}$). This observation agrees with the spectral properties of the preconditioners described in Section 6.2.

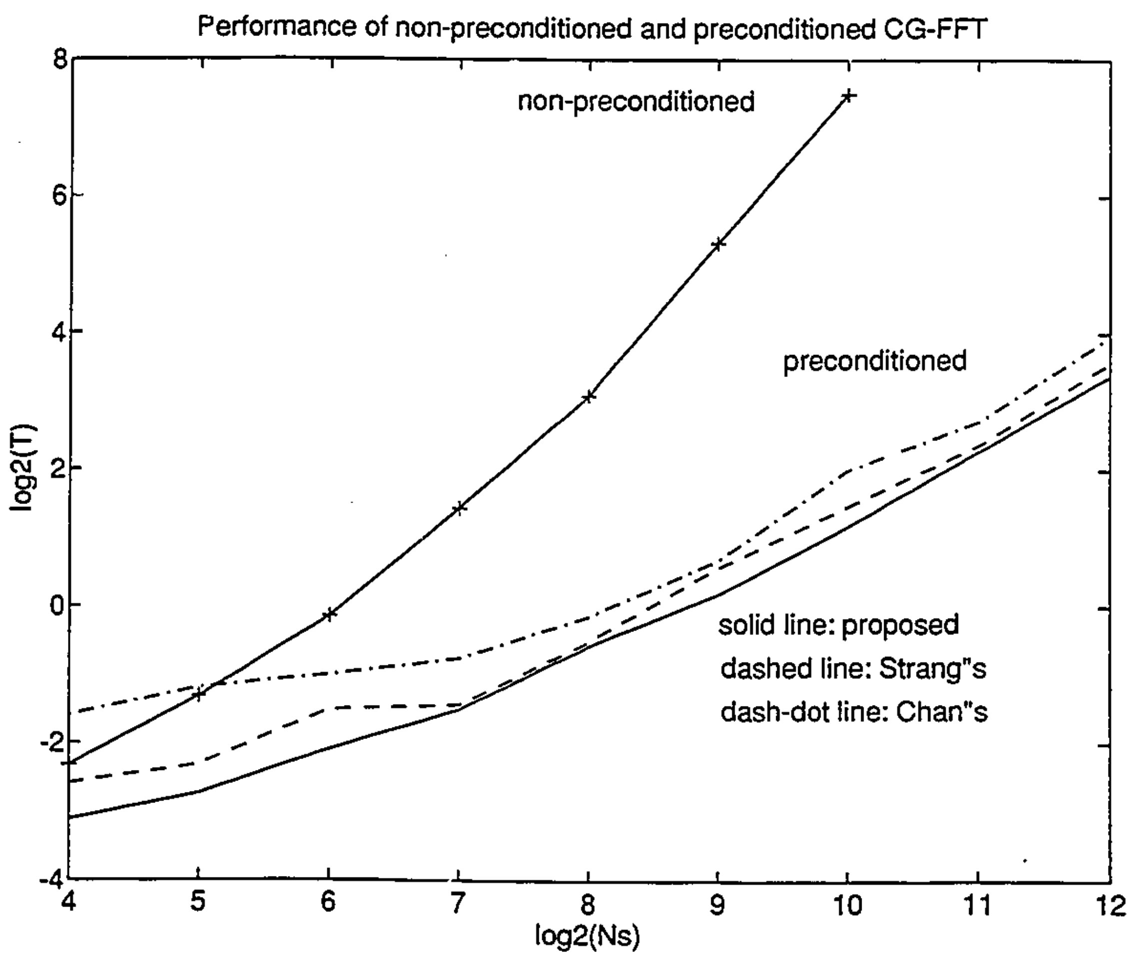


Figure 6.73: The log-log plot of the CPU time T vs. the number of unknowns N_s of Table 6.2. The asymptotic performance of the Strang's, Chan's and the proposed PCG-FFT algorithm can be determined from the gradients of the graphs.

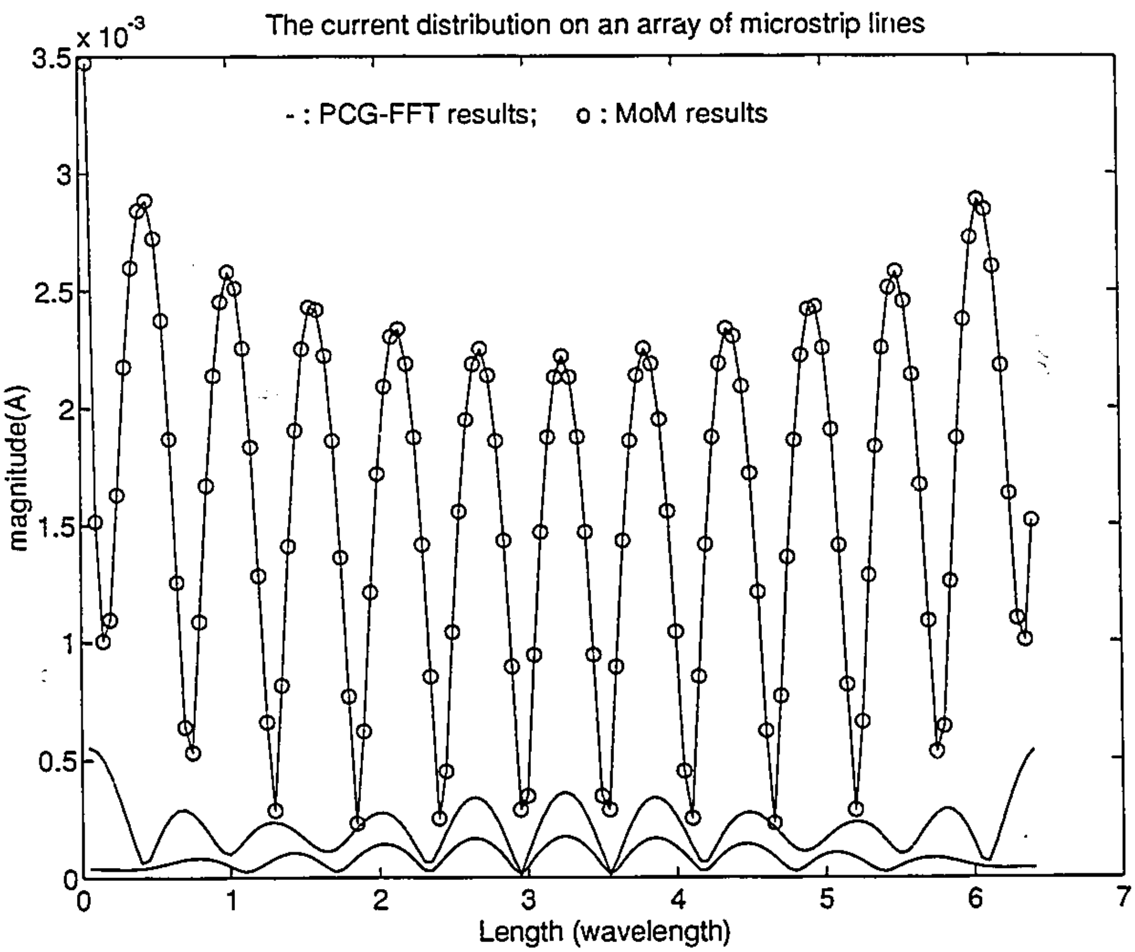


Figure 6.74: The current distribution on a microstrip line array

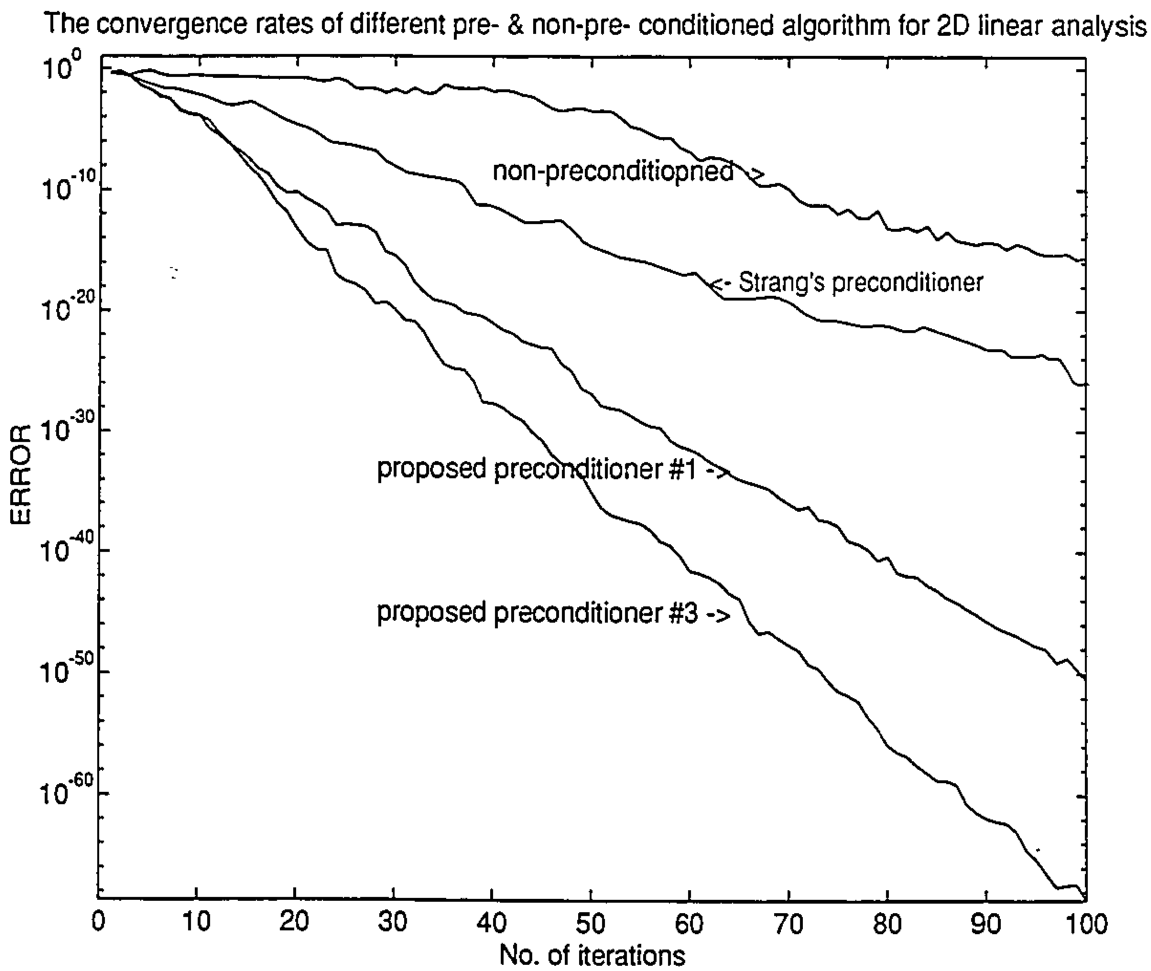


Figure 6.75: Comparison of rates of convergence for preconditioned and non-preconditioned algorithms for analyzing 2D linear problems

To use these techniques in a practical application, the simulation of the radiation from a microstrip dipole array (a type #2 hexagonal array with $M_1 = 8, M = 15$, and $N = 15$, see Fig.4.17 for parameters' definitions) has been carried out using the SPCG-FFT method. 18 basis functions were used to span a distance equal to a wavelength on each dipole. The number of unknowns for this problem is 5593. It takes 56 minutes for the CG-FFT to converge to an error of 10^{-8}, while it only takes less than 2 minutes for the SPCG-FFT to converge to the same error.

As a final example, a simple but large 2D problem was solved using both the CG-FFT and the SPCG-FFT. In this example, we derive the current distribution on a large rectangular metallic patch, which is excited by a plane wave. The angle of incidence for the wave is normal to the patch. The size of the patch is $10\lambda \times 10\lambda$. Figs.6.76 and 6.77 show the components of the surface current densities that are excited on the metallic patch. In this example, a total of 625 square pulse functions were used to sample the current on an area equal to $\lambda \times \lambda$. The current was sampled in two dimensions. The total number of the unknowns of this problem is 125,000. This large number of unknowns presents a great challenge in the implementation of the Method of Moments because of the requirement for a very large computer memory. In contrast, the CG-FFT can handle this problem with ease. On top of this, our SPCG-FFT technique enhances the convergence efficiency of the CG-FFT algorithm. This means that this iterative approach now possesses greater potential for solving large EM problems.

6.4 Summary

In this chapter, the spectrum and the convergence performance of the new highly efficient SPCG-FFT method have been studied.

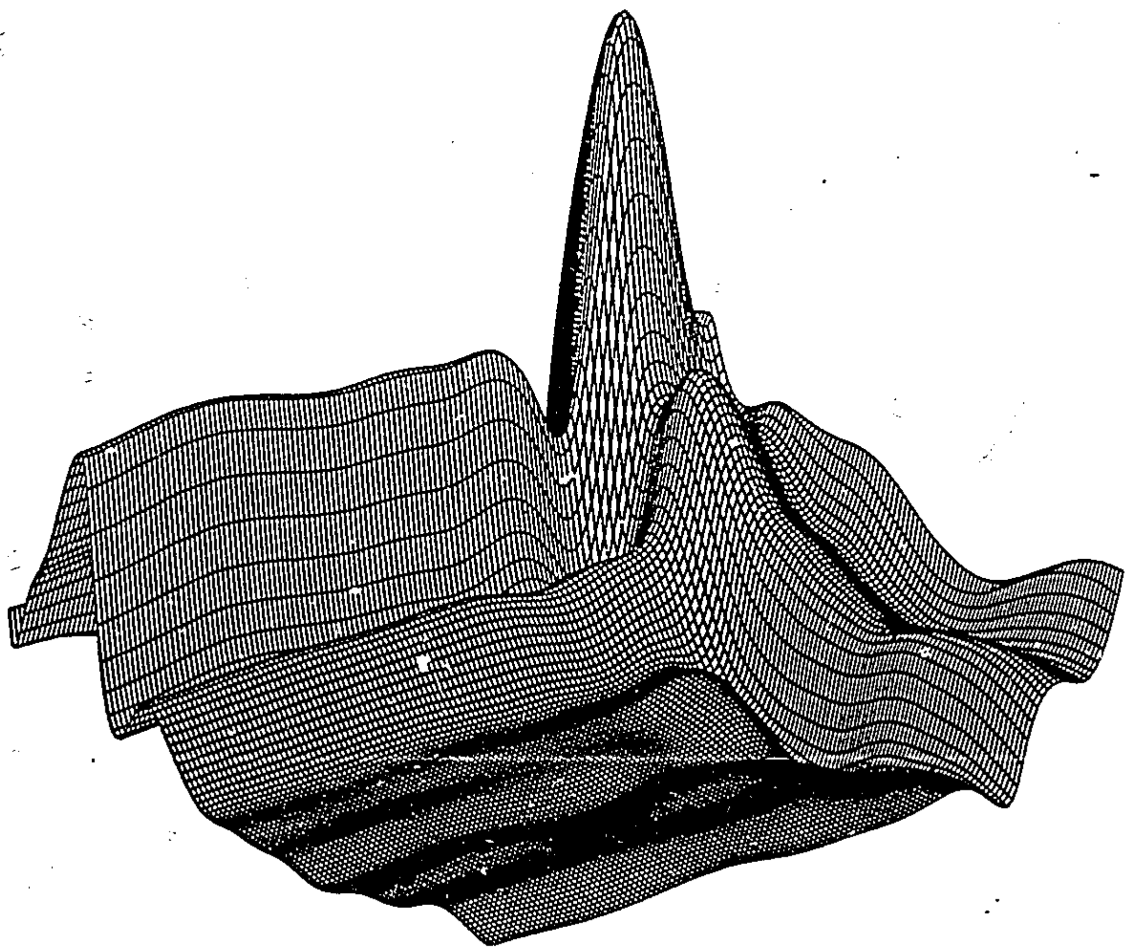


Figure 6.76: The induced co-polar current on a $10\lambda \times 10\lambda$ microstrip patch with a normally incident plane wave (only a symmetric quarter of the patch is shown on this graph)

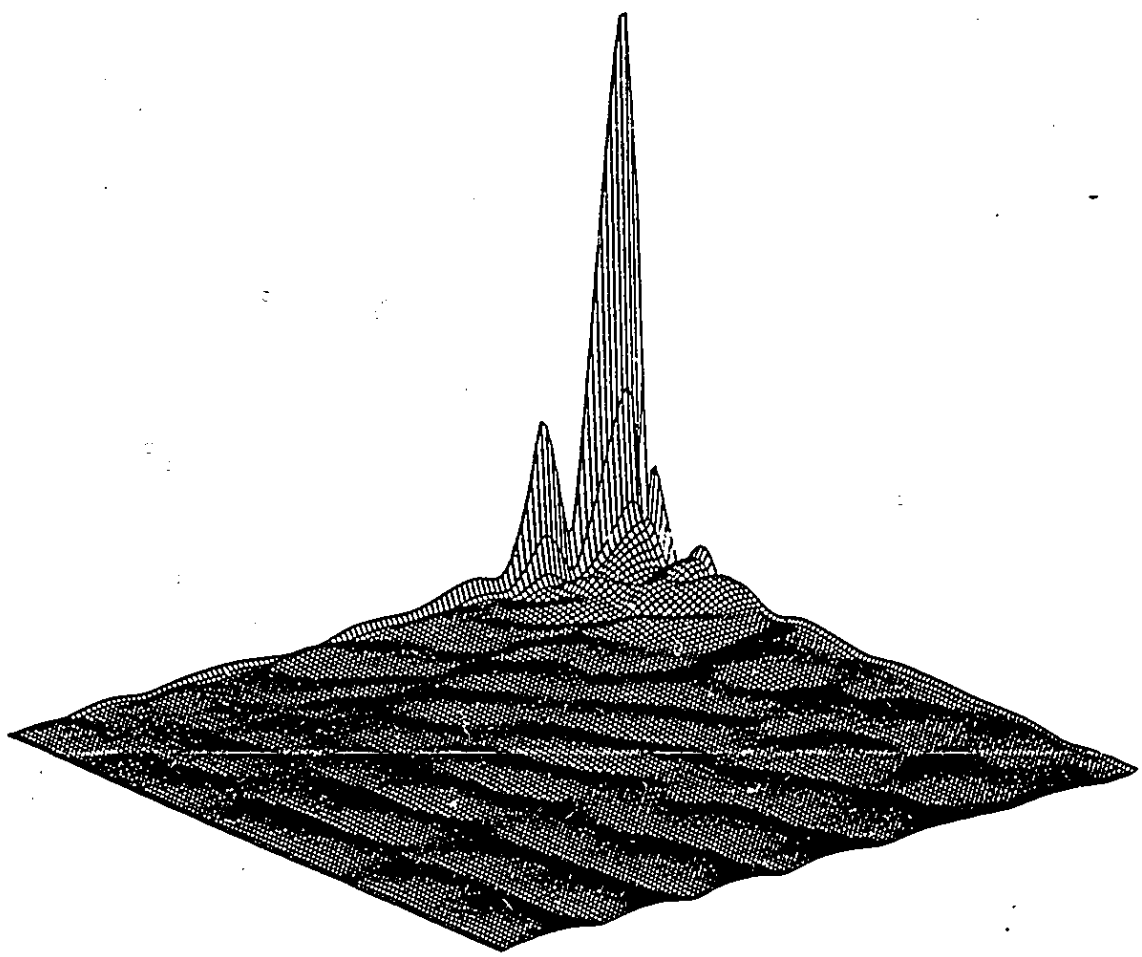


Figure 6.77: The induced cross-polar current on a $10\lambda \times 10\lambda$ microstrip patch with a normally incident plane wave (only a symmetric quarter of the patch is shown on this graph)

The study shows that for one-dimensional problems, this approach has similar performance to the Strang approach while it is much better than the Chan approach. However, for two-dimensional problems, our new method has been found to have superior spectral properties over all the other techniques.

The numerical results in this chapter have demonstrated that the new method can greatly improve the convergence rate of the conventional CG-FFT algorithms. Also, the new approach has superior convergence efficiency when compared with other preconditioning methods.

By means of numerical experiments, it has been found that the number of iterations required by the new method is independent of the number of the unknowns. The new method is found to converge with a fixed number of iterations, no matter how large the problem. This method achieves a computational complexity of $O(N_s \log N_s)$, in contrast to $O(N_i N_s \log N_s)$ for conventional CG-FFT algorithms.

Chapter 7

Conclusions

This thesis is devoted to the development of an accurate and efficient CAD tool that can handle very large and arbitrarily shaped arrays with feed networks. To reach the goal, a hybrid CG-FFT method and a highly efficient spectrally preconditioned CG-FFT method have been introduced in this thesis. Our efforts have been focused in two directions: (1) the improvement of the accuracy of the CG-FFT algorithm and (2) the improvement of the efficiency of this algorithm, both in terms of computer memory and computing time.

The thesis began with the development of a spatially discrete implementation scheme of the CG-FFT method. The distinguishing feature of this scheme was the treatment of the grad-div operator in the integral equation, combined with the spatially discrete procedure. This feature makes the algorithm very accurate because (1) no finite difference approximations are involved and (2) all the aliasing and truncation errors are eliminated. At the same time, the accuracy makes this algorithm very efficient because fewer iterations are needed to achieve the desired error criterion. The efficiency in memory size is achieved by decreasing the size of the zero padding required for implementing this algorithm.

In this thesis, the accurate and efficient spatially discrete scheme of the CG-FFT method has been extended to the analysis of microstrip problems. This extension was accomplished by developing a hybrid CG-FFT technique, which consists of a merging of the complex discrete image technique with the spatially discrete CG-FFT scheme. This combination supplies the full-wave closed-form Green's functions, which are essential for realizing the spatially discrete procedure, with negligible computational cost. Essentially, we avoid the cumbersome evaluation of the Sommerfeld integral, the latter being an inherent feature of microstrip problems. By implementing the spatially discrete scheme, we take advantage of all the algorithm's strength for solving free-space problems, and extend them to microstrip problems.

Another major accomplishment of this thesis is the development of a new highly efficient spectrally preconditioned CG-FFT method. A systematic approach for designing a class of spectral preconditioners for one-dimensional, linear two-dimensional and planar two-dimensional problems has been presented. The new method has been numerically tested. It has been demonstrated that the new technique has a superior convergence performance compared to that of the non-preconditioned CG-FFT algorithms. This method also has other advantages over existing preconditioning schemes, both in terms of memory efficiency and computational efficiency. In contrast to the computational complexity $O(N_i N_s \log N_s)$ of the CG-FFT method, the computational complexity $O(N_s \log N_s)$ has been achieved for the new method, which means that the number of the iterations required to achieve a solution with a certain error criterion is totally independent of the number of the unknowns involved, and the solution is obtained within a fixed number of iterations no matter how large the problem. All the numerical experiments have demonstrated the potential of this method, which suggests that the technique is worth further exploration.

The hybrid CG-FFT algorithm developed in this thesis has been adapted into a simulation software package to run on UNIX systems. The package includes a draw editor, an automatic mesh generator and a simulation engine. The software has been developed so that *What you draw is what you simulate* in a very user-friendly way.

Using this software, several types of microstrip antenna arrays have been numerically and experimentally studied. A number of features of interest to microstrip array designers, such as: edge effects, shape, size of an array, spurious radiation from the array's feed network, current distribution, input impedance, and the array's radiation pattern, have been analyzed in this thesis. Conclusions and discussions were presented. The capacity of the proposed hybrid full-wave CG-FFT method was demonstrated through modeling studies of very large microstrip reflectarrays. Design schemes and proposals for improvement were put forward, based on the the results of the modeling studies.

It should be mentioned that a number of assumptions were adopted in the algorithms developed in the thesis. It was assumed that the ground plane and the substrate of a microstrip structure were of infinite extent. The metallic surface of the microstrip antenna was assumed to be infinitely thin. These assumptions are not true for real microstrip structures, but it has been shown by previous researchers that these assumptions have little effect on the accuracy of the solutions. In actual fact, improvement can be made to reduce or eliminate some of these effects. For example, the method of treating the finite ground plane was proposed in [87]. The finite thickness of the metallic surface can be treated using the equivalent impedance boundary condition. All these improvements can be implemented with the new algorithm, but at a cost of increased computational complexity.

The microstrip structures dealt with in this thesis are limited to single layer planar structures. In fact, the method can be extended to a large range of microstrip problems, such as multilayer, CPW structures, and microstrip structures enclosed in metal boxes. The only thing that remains is the derivation of the closed-form Green's functions for each of these cases.

Bibliography

[1] J.R.James : " What's new in antennas?" *IEEE Antenna and Propagation Society Magazine,* Feb. 1990, pp.-18.

[2] D.M.Pozar:"Microstrip Antennas", *Proc. IEEE,* Vol.80, pp.9-91, Jan. 1992

[3] G.A. Deschamps : " Microstrip microwave antennas", Third U.S.A.F. Symp. on Antennas, 1953.

[4] K.R.Carver and J.W.Mink: " Microstrip antenna technology", *IEEE Trans. Antenna and Propagation,* Vol.AP-29, pp.-24, Jan. 1981

[5] W.F.Richards, Y.T.Lo and D.Harrison: " An improved theory for microstrip antennas and applications", *IEEE Trans. Antenna and Propagation,* Vol.AP-29, pp.8-46, Jan. 1981.

[6] J.R.James and P.S.Hall, *Handbook of microstrip Antennas*, London, U.K.: Peter Peregrinus, 1989.

[7] K.S.Yee: " Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media", *IEEE Trans. Antenna and Propagation,* Vol.AP-17, pp.85-589, May 1966.

[8] I.E.Rana and N.G.Alexopolous: " Current distribution and input impedance of printed dipoles", *IEEE Trans. Antenna and Propagation,* Vol.AP-29, pp.9-105, Jan. 1981.

[9] J.R.Mosig and F.E.Gardiol: " General integral equation formulations for microstrip antenna and scatters", *Proc. IEE Pt.H,* Vol-132, pp.24-432, 1985.

[10] D.M.Pozar: " Input impedance and mutual coupling of rectangular microstrip antennas", *IEEE Trans. Antenna and Propagation,* Vol.AP-30, pp.191-1196, Nov. 1982.

[11] C.Y.Shen, K,J,Glover, M.I.Sancer, and A.D.Varvatsis: " The discrete Fourier transform method of solving differential-integral equations in scattering theory", *IEEE Trans. Antenna and Propagation,* Vol.AP-37, pp.032-1041, Aug. 1989.

[12] R.Kastner, E,Heyman and A.Sabban: " Spectral domain iterative analysis of single and double-layered microstrip antennas using the conjugate gradient algorithm", *IEEE Trans. Antenna and Propagation,* Vol.AP-36, pp.204-1212, 1988.

[13] T.M.Willis, P.L.Sengupta: " Spectral analysis of microstrip antennas with CG-FFT : two-dimensional results", *IEEE Trans. Antenna and Propagation.* Vol.AP-37, pp.10-816, July 1989.

[14] A.F.Peterson, 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. Antenna and Propagation,* Vol.AP-36, pp.177-1179, Aug. 1988

[15] A.F.Perterson and R.Mittra: " Convergence of the conjugate gradient method when applied to matrix equations representing electromagnetic scattering

problems", *IEEE Trans. Antenna and Propagation*, Vol.AP-34, pp.477-1454, Dec. 1986.

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

[17] T.K.Sakar, E. Arvas & S.S.Rao : " Application of FFT and Conjugate Gradient Method for the solution of electromagnetic radiation from electrically large and small conducting bodies " IEEE Trans. AP, 1986, AP-34, pp.635-640

[18] T.K.Sarkar, " The application of the conjugate gradient method for the solution of operator equations arising in electromagnetic scattering from wire antennas ", *Radio Science*, Vol.19, pp. 1156-1172, Sept.-Oct., 1984

[19] M.F.Catedra, R.P.Torres, J.G.Guevas: " A method to analyze scattering from general periodic screens using fast Fourier transform and conjugate gradient method", *Electromagnetic Modeling and Measurements for Analysis and Synthesis Problems*, B.d.Neumann, ed., NATO ASI Series, Series E : Applied Sciences - Vol.199, 1991.

[20] N.N.Bojarski, " k-space formulation of the electromagnetic scattering problem ", *Air Force Avionics Lab. Technical Report* , AFAL-TR-71-75, March 1971

[21] J.M.Jin and J.L.Volakis, " A biconjugate gradient FFT solution for scattering by planar plates", *Electromagnetics*, Vol.?, pp. ??-??, Mar. 1992.

[22] Yuan Zhuang, Ke-Li Wu, Chen Wu, John Litva : " A combined full-wave CG-FFT method for rigorous analysis of large microstrip antenna arrays". *IEEE Trans. Antennas and Propagation*, Vol.AP-44, January, 1996.

[23] Yuan Zhuang, Ke-Li Wu, John Litva : " Application of CG-FFT Combined with Discrete Image Technique in the analysis of microstrip antennas" IEEE AP-S & URSI Joint Symp. Dig., 1992

[24] D.G.Fang, J.J.Yang, and G.Y.Delisle, " Discrete image theory for horizontal electric dipoles in a multilayered medium", *IEE Proc.H, Microwaves, Antennas and Propagation*, Vol.135, pp. 297-303, 1988

[25] T.J.Peters and J.L.Volakis: " Application of a conjugate gradient FFT method to scattering from thin planar material plates", *IEEE Trans. Antenna and Propagation,* Vol.AP-36, pp.18-526, April 1988.

[26] A.P.M.Zwamborn and P.M.van den Berg, " A weak form of the conjugate gradient FFT method for plate problems ", *IEEE Trans. Antennas and Propagation*, Vol. AP-39, Feb. 1991

[27] D.T.Borup and O.P.Gandhi: " Calculation of high resolution SAR distribution in biological bodies using the FFT algorithm and the conjugate gradient method", *IEEE Trans. Microwave Theory and Technology,* Vol.AP-33, pp.17-419, May 1985.

[28] R.Kastner and R.Mittra: " Iterative analysis of finite sized planar frequency selective surface with rectangular patches or perforations", *IEEE Trans. Antenna and Propagation,* Vol.AP-35, pp.72-377, April 1987.

[29] T.A.Cwik and R.Mittra: "Scattering from a periodic array of free-standing arbitrary shaped perfectly conducting or resistive patches", *IEEE Trans. Antenna and Propagation,* Vol.AP-35, pp.226-1234, Nov 1987.

[30] K.Barkeshli and J.L.Volakis: " Improving the convergence rate of the conjugate gradient FFT method using subdomain basis functions", *IEEE Trans. Antenna and Propagation,* Vol.AP-37, pp.93-900, July 1989.

[31] P.Petre: " Applications of the generalized biconjugate gradient FFT method for analyzing arrays of resistive patches standing in free space and in front of a ground plane covered by a dielectric layer", in Proc. 1989, URSI. Int. Symp., Electromagnetic Theory, Stockholm, Sweden, Aug. 14-18, 1989, pp.38-340.

[32] M.F.Catedra, J. G. Cuevas and L.Nuño " A scheme to analyze conducting plates of resonant size using the conjugate gradient method and the fast Fourier transform", *IEEE Trans. Antenna and Propagation,* Vol.AP-36, pp.1744-1752, Dec. 1988.

[33] M.F.Catedra and E.Gago, " Spectral Domain analysis of conducting patches of arbitrary geometry in multilayer media using the CG-FFT method" , *IEEE Trans. Antenna and Propagation*, Vol. AP-38, Oct. 1990.

[34] Enrique Drake, Frnacisco Medina, Manual Horno: " An improved iterative technique for the quasi-TEM analysis of generalized planar lines", *IEEE Trans. Microwave Theory and Technology,* Vol.AP-40, pp.652-658, 1992.

[35] Y.L.Chow, J.J.Yang, D.G.Fang, and G.E.Howard, " Closed-form spatial Green's function for thick substrate", *IEEE Trans. Microwave Theory and Technology*, Vol.39, pp. 588-592, Mar. 1991.

[36] R.E.Collin: *Field Theory of Guided Waves*, New York, U.S.A. IEEE Press, 1991.

[37] Chen Wu, Ke-Li Wu, Zhi-Qiang Bi, and John Litva, " Accurate characterization of planar printed antennas using FDTD method", *IEEE Trans. Antennas and propagation*, Vol. AP-40, May 1992.

[38] P.B.Katehi and N.G.Alexopoulos: " On the modeling of electromagnetically coupled microstrip antennas - the printed strip dipole", *IEEE Trans. Antenna and Propagation,* Vol.AP-32, pp.179-1186, Nov. 1984.

[39] D.M.Pozar: "Input impedance and mutual coupling of rectangular microstrip antennas", *IEEE Trans. Antenna and Propagation,* Vol.AP-30, pp.191-1196, Nov. 1982.

[40] D.M.Pozar: " Consideration for millimeter wave printed antennas", *IEEE Trans. Antenna and Propagation,* Vol.AP-31, pp.40-747, 1983

[41] Y.T.Lo, D.Solomon and W.F.Richards: " Theory and experiment on microstrip antennas", *IEEE Trans. Antenna and Propagation,* Vol.AP-27, pp.37-145, 1979.

[42] C.Wood: " Improved bandwidth of microstrip antennas using parasitic elements", *Proc. IEE Pt.H,* Vol-127, pp.31-234, Aug. 1984.

[43] K.-L.Wu, J.Litva, R.Fralich and C.Wu: " Full wave analysis of arbitrary shaped line-fed microstrip antenna using triangular finite element method", *Proc. IEE Pt.H,* Vol-, pp.21-427, Oct. 1991.

[44] C.C.Liu, J. Shmoys, and A.Hessel: " E-plane performance trade-off in two-dimensional microstrip patch element phased array", *IEEE Trans. Antenna and Propagation,* Vol.AP-30, pp.201-1206, 1982

[45] D.M.Pozar, D.H.Schaubert: " Scan blindness in infinite phased arrays of printed dipoles", *IEEE Trans. Antenna and Propagation*, Vol.AP-32, pp.02-610, June 1984.

[46] D.M.Pozar: " Analysis of an infinite array of rectangular microstrip patches with idealized probe feeds", *IEEE Trans. Antenna and Propagation*, Vol.AP-32, pp.101-1107, Oct. 1984.

[47] D.M.Pozar: " Scanning characteristic of infinite arrays of printed antenna subarrays", *IEEE Trans. Antenna and Propagation*, Vol.AP-40, pp.66-674, 1992.

[48] R.C.Hansen: " Evaluation of the large array method", *Proc. IEE Pt.H*, Vol-137, pp.4-98, 1990.

[49] M.D.Deshpande and M.C.Bailey: " Analysis of finite phase arrays of circular microstrip patches", *IEEE Trans. Antenna and Propagation*, Vol.AP-37, pp.355-1360. Nov. 1989.

[50] D.M.Pozar: " Analysis of finite phased arrays of printed dipoles", *IEEE Trans. Antenna and Propagation*, Vol.AP-33, pp.045-1053, Oct. 1985.

[51] A.K.Bhattacharyya and L.Shafai: " Surface wave coupling between circular patch antennas", *Electronic Letters*, Vol.22, pp.198-1200, Oct. 1986.

[52] V.Hansen: " Finite array of printed dipoles with a dielectric cover", *Proc. IEE Pt.H*, Vol-134, pp.61-269, June 1987.

[53] Ke-Li Wu, M. Spenuk, John Litva, D.-G. Fang, " Theoretical and experimental study of feed network effects on the radiation pattern of series-fed microstrip antenna arrays ", *IEE Proc.H, Microwaves, Antennas and Propagation*, Vol.138, June 1991.

[54] Ke-Li Wu, Yuan Zhuang, Chen Wu and John Litva: " An efficient full-wave analysis technique for large microstrip antenna arrays" *International Symposium of Computational Electromagnetics*, 1993.

[55] R.C.Hansen (Ed.), *Microwave Scanning Antennas*, Vol.2, 400pp.Academic, Orland, Fla., 1965.

[56] L.Stark: " Radiation impedance of a dipole in a infinite planar phased array", *Radio Science*, 7, pp.61-377, 1966.

[57] P.S.Hall, C.M.Hall: " Coplanar corporate feed effects in microstrip patch array design", *Proc. IEE Pt.H,* Vol-135, pp.80-186, June 1988.

[58] L.Lewin: " Spurious radiation from microstrip", *Proc. IEE Pt.H,* Vol-125, pp.33-642, July 1978.

[59] E.Levine, Malamud, S.Shtrikman and D.Treves: " A study of microstrip array antennas with feed networks", *IEEE Trans. Antenna and Propagation,* Vol.AP-37, pp.26-434, 1989.

[60] H.Y.Yang, N.G.Alexopoulos: " Generation of nearly hemispherical and high gain azimuthally symmetric patterns with printed circuit antennas", *IEEE Trans. Antenna and Propagation,* Vol.AP-35, pp.72-977, 1987.

[61] J.Huang, A.Densmore, A.Tulintseff, and V.Jamnejad, "Downsizing antenna technology for mobile and satellite communications " IEEE AP-S & URSI Joint Symp. Dig., 1993, pp.1834-1837

[62] R.E.Munson, H.H.Haddad and J.Hanlen: " Microstrip reflectarray for satellite communication and RCS enhancement or reduction", U.S.Patent 4,684,952, Aug 1987.

[63] Chang,D.C.,Huang,M.C. : " Microstrip reflectarray antenna with offset feed " Electronic Letters, 1992, pp.1489-1491

[64] John Huang " Microstrip Reflectarray " IEEE AP-S Int. Symp. Dig., 1991, pp.612-615

[65] D.M.Pozar and T.A.Metzler, "Analysis of a reflectarray antenna using microstrip patches of variable size" Electronic Letters, 1993,

[66] T.A.Metzler, "Design and analysis of a microstrip reflectarray", Ph.D. dissertation, Univ. of Massachusetts at Amherst, 1993.

[67] Yuan Zhuang, Ke-Li Wu, Chen Wu, and John Litva, " Microstrip reflectarray: full-wave analysis and design scheme", *1993 IEEE International Symposium on Antennas and Propagation* .

[68] Yuan Zhuang, John Litva and Chen Wu, " Theoretical and experimental study of microstrip reflectarray used in mobile satellite communications", International Antenna Applications Symposium, Illinois, Sept. 1993.

[69] Yuan Zhuang, John Litva, Chen Wu and Ke-Li Wu, " Modeling study of microstrip reflectarray", *IEE Proc.H, Microwaves, Antennas and Propagation,* Vol.142, February 1995, pp.78-80

[70] C.Lancozos: " An iterative method for the solution of the eigenvalue problem of linear differential and integral operators", *J.Res. Nat. Bur. Stand.,* Vol.45, pp.55-282, 1950.

[71] C.Lancozos: " Solution of systems of linear equations by minimized iterations", *J.Res. Nat. Bur. Stand.,* Vol.49, pp.3-53, 1952.

[72] T.K.Sakar: " On the application of the generalized biconjugate gradient method", *J. Electromagn. Waves and App.*, Vol.1, No.3, pp.23-242, 1987.

[73] C.F.Smith, A.F.Peterson and R.Mittra: " The biconjugate gradient method for electromagnetic scattering", *IEEE Trans. Antenna and Propagation,* Vol.AP-38, pp.38-940, 1990.

[74] S.C.Eisenstat, H.C.Elman, and M.H.Schultz: " Variational iterative methods for nonsymmetric systems of linear equations", *SIAM J. Numer. Anal.*, Vol.20, pp.345-357, April 1983

[75] F.X.Canning: " Physical and mathematical structure determine convergence rate of iterative techniques", *IEEE Trans. Magnetic,* Vol.25, pp.2825-2827, July 1989

[76] A.Jennings: " Influence of the eigenvalue spectrum on the convergence of the conjugate gradient method", *J. Inst. Math. Applications,* Vol.20, pp.1-72, 1977.

[77] H.C.Elman: " Iterative methods for large , sparse, non-symmetric systems of linear equations", *Research Report,* No.229, Dept. of Computer Science, Yale Univ., New Haven, CT., April 1982.

[78] D.J.Evan (Ed.): *Preconditioning methods: theory and applications*, New York: Gorden and Breach Science, 1983.

[79] A.P.Peirce, S.Spottiswood, and J.A.L.Napier: " The spectral boundary element method : a new window on boundary elements in rock mechanics", *Int. J. Rock Mech. Min. Sci. & Geomech. Abstr.*, Vol.29, pp.379-400, 1992

[80] T.J.Judasz and B.B.Balsley: " Improved theoretical and experimental models for the coaxial collinear antennas", *IEEE Trans. Antenna and Propagation*, Vol.AP-37, pp.89-296, March, 1989.

[81] Arnold Kas and Elizabeth L. Yip: " Preconditioned Conjugate Gradient Methods for solving electromagnetic problems", *IEEE Trans. Antenna and Propagation*, Vol.AP-35, pp.47-152, Feb. 1987.

[82] C.F.Smith: " The performance of preconditioned iterative methods in computational electromagnetics", Ph.D. dissertation, Univ. of Illinois at Urbana-Champaign, 1987.

[83] A.V.Oppheim and R.W.Schafer: *Digital Signal Processing*, Englewood Cliff, NJ: Pretice-Hall, 1975.

[84] G.Strang: " A proposal for *Toeplitz* matrix calculations", *Studies in App. Math.*, Vol.74, pp.71-176, 1986.

[85] T.F.Chan: " An optimal circulant preconditioner for *Toeplitz* systems", *SIAM J. Sci. Stat. Comput.* , Vol.9, pp.66-771, July 1988.

[86] P.Davis: *Circulant Matrices*, New York: Wiley, 1979.

[87] S.A.Bokhari, J.R.Mosig and F.E.Gardiol: " Radiation pattern computation of microstrip antennas on finite size ground plane", *Proc. IEE Pt.H*, Vol-139, pp.278-286, June 1992

Appendix A

Formulations of Fourier Transform and Discrete Fourier Transform

The one- and two-dimensional Fourier transforms of a function $g(x)$ and $g(x,y)$ are defined as,

$$
\begin{aligned}
1D &: \\
\mathcal{F}[g] &= \tilde{g}(k_x) = \int_{-\infty}^{\infty} g(x) e^{-jk_x x} dx && \text{(A.243)} \\
\mathcal{F}^{-1}[\tilde{g}] &= g(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{g}(k_x) e^{jk_x x} dk_x && \text{(A.244)} \\
2D &: \\
\mathcal{F}[g] &= \tilde{g}(k_x, k_y) = \int_{-\infty}^{\infty} g(x,y) e_{-j(k_x x + k_y y)} dx dy && \text{(A.245)} \\
\mathcal{F}^{-1}[\tilde{g}] &= g(x,y) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{g}(k_x, k_y) e^{j(k_x x + k_y y)} dk_x dk_y. && \text{(A.246)}
\end{aligned}
$$

The convolutions are defined as

$$
1D: \qquad g(x) * h(x) \qquad = \int_{-\infty}^{\infty} g(\tau) h(x - \tau) d\tau \qquad \text{(A.247)}
$$

$$2D: \quad g(x,y) * h(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(\tau,\zeta)h(x-\tau, y-\zeta)d\tau d\zeta. \quad (A.248)$$

The convolution theorem says

$$\mathcal{F}[g(x) * h(x)] = \tilde{g}(k_x)\cdot\tilde{h}_(k_x) \quad (A.249)$$

$$\mathcal{F}[g(x,y) * h(x,y)] = \tilde{g}(k_x,k_y)\cdot\tilde{h}(k_x,k_y). \quad (A.250)$$

In the computer solutions, continuous functions must be replaced by discrete versions of these functions. Therefore, the continuous Fourier transform must be replaced with the discrete Fourier transform (DFT) in order to be able to make use of the efficient fast Fourier transform (FFT).

If a finite duration continuous function $h(x)$ is sampled at a space interval Δx, the resulted finite sequence is

$$h(n) = h(x_n); \quad x_n = n\Delta x, \quad n = 0, 1, ..., N-1. \quad (A.251)$$

The quantity $\frac{1}{\Delta x}$ is called the sampling rate. The discrete Fourier transforms of this finite sequence are

$$H(k) = \mathcal{DFT}[h(n)] = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}nk} \quad (0 \le k \le N-1) \quad (A.252)$$

$$h(n) = \mathcal{IDFT}[H(k)] = \frac{1}{N}\sum_{k=0}^{N-1} H(k)e^{j\frac{2\pi}{N}nk} \quad (0 \le n \le N-1). \quad (A.253)$$

Similarly, 2D DFTs are defined as

$$H(k,l) = \mathcal{DFT}[h(m,n)] = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} x(m,n)e^{-j(\frac{2\pi}{N}nl+\frac{2\pi}{M}mk)}$$
$$(0 \le k \le M-1; o \le l \le N-1) \quad (A.254)$$

$$h(m,n) = \mathcal{IDFT}[H(k,l)] = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=1}^{N-1} H(k,l)e^{j(\frac{2\pi}{M}mk+\frac{2\pi}{N}nl)}$$
$$(0 \le m \le M-1; 0 \le n \le N-1). \quad (A.255)$$

The relation between the Fourier transform and discrete Fourier transform can be described in the following way. That is, if $h(n)$ is the discrete form of a finite duration continuous function $h(x)$, we can generate a periodic sequence

$$h_p(n) = h(n \ [\text{mod } N]) \tag{A.256}$$

i.e.

$$h(n) = h_p(n)\mathcal{R}_N(n) \tag{A.257}$$

where

$$\mathcal{R}_N(n) = \begin{cases} 1, & 0 \leq n \leq N-1 \\ 0, & elsewhere \end{cases} . \tag{A.258}$$

Then

$$\mathcal{DFT}[h(n)] = \mathcal{F}[h_p(n)]. \tag{A.259}$$

The next question is the accuracy that is achieved when we replace a continuous FT by its corresponding DFT. The sampling theorem can answer this question.

Sampling Theorem *For a band-width limited function $h(x)$, i.e.,$H(k) = 0$ for all $|k| > k_m$, the information content of $h(x)$ can be 100% recovered from its discrete form $h(n)$ if the sampling rate $\Delta \mathrm{x} \leq \frac{1}{2\mathrm{k_m}}$. This sampling rate is called Nyquist rate and k_m is called Nyquist frequency.*

However, if the continuous function is not band-width limited to the Nyquist frequency, it turns out that all of the spectrum which lies outside of the frequency range $-k_m < k < k_m$ is aliased into the range by the very act of discrete sampling. This phenomena is called *aliasing*. Thereby, we have to be very careful when using FFTs to replace continuous FTs.

Appendix B

The Lists of Eigenvalues and Condition Numbers

Table B.1 the eigenvalues and condition numbers of A and the preconditioned matices

A		$Z^{-1}A$		$S^{-1}A$		$C^{-1}A$	
real	imag	real	imag	real	imag	real	imag
26.481	50.971	1.613	-0.696	1.692	-0.692	1.195	-1.292
25.886	49.979	0.383	0.080	0.389	0.077	0.550	0.571
24.896	48.316	0.849	0.291	0.863	0.240	0.310	0.069
23.513	45.969	0.878	0.083	0.903	0.062	1.380	0.221
21.743	42.915	1.028	0.005	1.035	0.007	1.264	0.151
19.589	39.125	0.971	-0.004	0.972	-0.001	0.762	0.012
17.061	34.550	1.009	0.007	1.025	-0.001	1.171	0.071
14.173	29.126	0.992	-0.006	1.018	-0.002	0.866	-0.037
10.952	22.739	0.999	0.003	0.994	-0.009	1.149	0.062
7.452	15.213	0.999	0.003	1.003	0.010	0.890	-0.002
3.953	6.108	0.998	0.001	0.999	-0.006	1.124	0.046
1.587	-4.566	0.998	0.001	1.000	0.005	1.114	0.042
0.803	-15.385	0.999	0.001	1.003	0.001	1.104	0.036
0.570	-25.693	0.999	0.001	1.000	-0.003	1.100	0.035
0.429	-35.855	1.000	0.000	1.000	0.003	1.096	0.032
0.347	-45.992	0.999	0.001	0.998	-0.002	1.094	0.032
0.284	-56.178	1.000	0.001	0.999	-0.002	0.936	-0.013
0.241	-66.428	1.000	0.000	1.000	-0.002	0.941	-0.004
0.206	-76.760	1.000	0.000	1.000	0.002	0.961	-0.007
0.179	-87.171	1.000	0.000	1.001	0.001	0.963	-0.002
0.157	-97.661	1.000	0.000	1.001	0.000	0.971	-0.005
0.139	-108.219	1.000	0.000	1.000	0.000	0.975	-0.002
0.123	-118.840	1.000	0.000	1.000	0.000	0.975	-0.001

A		$Z^{-1}A$		$S^{-1}A$		$C^{-1}A$	
real	imag	real	imag	real	imag	real	imag
0.111	-129.510	1.000	0.000	1.000	0.000	1.027	0.001
0.099	-140.221	1.000	0.000	1.000	0.000	0.982	-0.002
0.090	-150.958	1.000	0.000	1.000	0.000	0.982	-0.001
0.082	-161.713	1.000	0.000	1.000	0.000	0.987	-0.001
0.074	-172.471	1.000	0.000	1.000	0.000	0.987	-0.001
0.068	-183.221	1.000	0.000	1.000	0.000	0.990	-0.001
0.062	-193.951	1.000	0.000	1.000	0.000	0.991	-0.001
0.057	-204.650	1.000	0.000	1.000	0.000	0.993	-0.001
0.052	-215.305	1.000	0.000	1.000	0.000	0.993	0.000
0.048	-225.906	1.000	0.000	1.000	0.000	0.995	-0.001
0.045	-236.441	1.000	0.000	1.000	0.000	0.995	0.000
0.041	-246.900	1.000	0.000	1.000	0.000	0.997	-0.001
0.038	-257.272	1.000	0.000	1.000	0.000	0.997	0.000
0.035	-267.547	1.000	0.000	1.000	0.000	0.998	-0.001
0.033	-277.715	1.000	0.000	1.000	0.000	0.998	0.000
0.030	-287.768	1.000	0.000	1.000	0.000	0.999	-0.001
0.028	-297.696	1.000	0.000	1.000	0.000	0.999	0.000
0.026	-307.491	1.000	0.000	1.000	0.000	1.000	0.000
0.024	-317.143	1.000	0.000	1.000	0.000	1.000	0.000
0.023	-326.647	1.000	0.000	1.000	0.000	1.001	-0.001
0.021	-335.993	1.000	0.000	1.000	0.000	1.001	0.000
0.020	-345.176	1.000	0.000	1.000	0.000	1.001	0.000
0.018	-354.189	1.000	0.000	1.000	0.000	1.001	0.000
0.017	-363.024	1.000	0.000	1.000	0.000	1.001	0.000

A		$Z^{-1}A$		$S^{-1}A$		$C^{-1}A$	
real	imag	real	imag	real	imag	real	imag
0.016	-371.677	1.000	0.000	1.000	0.000	1.001	0.000
0.015	-380.143	1.000	0.000	1.000	0.000	1.001	0.000
0.014	-388.415	1.000	0.000	1.000	0.000	1.001	0.000
0.013	-396.491	1.000	0.000	1.000	0.000	1.001	0.000
0.012	-404.364	1.000	0.000	1.000	0.000	1.001	0.000
0.011	-412.033	1.000	0.000	1.000	0.000	1.001	0.000
0.010	-419.493	1.000	0.000	1.000	0.000	1.001	0.000
0.010	-426.741	1.000	0.000	1.000	0.000	1.001	0.000
0.009	-433.775	1.000	0.000	1.000	0.000	1.001	0.000
0.009	-440.593	1.000	0.000	1.000	0.000	1.001	0.000
0.008	-447.192	1.000	0.000	1.000	0.000	1.001	0.000
0.007	-453.573	1.000	0.000	1.000	0.000	1.001	0.000
0.007	-459.732	1.000	0.000	1.000	0.000	1.001	0.000
0.006	-465.670	1.000	0.000	1.000	0.000	1.001	0.000
0.006	-471.387	1.000	0.000	1.000	0.000	1.001	0.000
0.005	-476.882	1.000	0.000	1.000	0.000	1.002	0.000
0.005	-482.156	1.000	0.000	1.000	0.000	1.002	0.000
0.005	-487.210	1.000	0.000	1.000	0.000	1.002	0.000
0.004	-492.044	1.000	0.000	1.000	0.000	1.002	0.000
0.004	-496.661	1.000	0.000	1.000	0.000	1.002	0.000
0.004	-501.060	1.000	0.000	1.000	0.000	1.002	0.000
0.003	-505.246	1.000	0.000	1.000	0.000	1.002	0.000
0.003	-509.220	1.000	0.000	1.000	0.000	1.002	0.000
0.003	-512.983	1.000	0.000	1.000	0.000	1.002	0.000

A		$Z^{-1}A$		$S^{-1}A$		$C^{-1}A$	
real	imag	real	imag	real	imag	real	imag
0.003	-516.540	1.000	0.000	1.000	0.000	1.002	0.000
0.002	-519.894	1.000	0.000	1.000	0.000	1.002	0.000
0.002	-523.046	1.000	0.000	1.000	0.000	1.002	0.000
0.002	-526.002	1.000	0.000	1.000	0.000	1.002	0.000
0.002	-528.763	1.000	0.000	1.000	0.000	1.002	0.000
0.002	-531.333	1.000	0.000	1.000	0.000	1.002	0.000
0.002	-533.716	1.000	0.000	1.000	0.000	1.002	0.000
0.001	-535.912	1.000	0.000	1.000	0.000	1.002	0.000
0.001	-537.915	1.000	0.000	1.000	0.000	1.002	0.000
0.000	-539.348	1.000	0.000	1.000	0.000	1.002	0.000
0.000	-539.407	1.000	0.000	1.000	0.000	1.002	0.000
0.000	-539.601	1.000	0.000	1.000	0.000	1.002	0.000
0.000	-539.527	1.000	0.000	1.000	0.000	1.002	0.000
0.000	-539.774	1.000	0.000	1.000	0.000	1.002	0.000
0.000	-539.878	1.000	0.000	1.000	0.000	1.002	0.000
0.000	-540.162	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-540.384	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-540.994	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-540.744	1.000	0.000	1.000	0.000	1.003	0.000
0.001	-541.456	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-541.465	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-541.989	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-542.276	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-542.951	1.000	0.000	1.000	0.000	1.003	0.000

A		$Z^{-1}A$		$S^{-1}A$		$C^{-1}A$	
real	imag	real	imag	real	imag	real	imag
0.000	-542.838	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-543.493	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-543.760	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-544.457	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-544.400	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-545.086	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-545.331	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-545.934	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-546.030	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-546.819	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-546.700	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-547.402	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-547.575	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-548.145	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-548.232	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.865	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.865	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.745	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.745	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.545	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.547	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.265	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-551.275	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-550.933	1.000	0.000	1.000	0.000	1.003	0.000

A		$Z^{-1}A$		$S^{-1}A$		$C^{-1}A$	
real	imag	real	imag	real	imag	real	imag
0.000	-550.906	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-550.524	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-550.471	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-550.048	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-549.971	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-549.497	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-548.860	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-548.842	1.000	0.000	1.000	0.000	1.003	0.000
0.000	-549.422	1.000	0.000	1.000	0.000	1.003	0.000
CONDITION NUMBER:							
116.4918		38.3810		39.8722		34.6112	

Table B.2 Eigenvalues and condition numbers of four proposed preconditioned matrices

$(Z_p^{(I)})^{-1} A$		$(Z_p^{(-I)})^{-1} A$		$(Z_p^{(++)})^{-1} A$		$(Z_p^{(+-)})^{-1} A$	
real	imag	real	imag	real	imag	real	imag
1.613	-0.696	-0.950	-0.997	-0.950	-0.997	0.383	0.080
0.383	0.080	0.661	0.101	1.613	-0.696	0.922	-0.352
0.849	0.291	0.922	-0.352	0.849	0.291	0.661	0.101
0.878	0.083	1.130	-0.139	1.130	-0.139	0.878	0.083
1.028	0.005	1.030	0.004	0.971	-0.004	1.030	0.004
0.971	-0.004	0.973	-0.004	0.973	-0.004	1.028	0.005
1.009	0.007	0.991	-0.007	0.991	-0.007	1.009	0.007
0.992	-0.006	1.008	0.007	0.992	-0.006	1.008	0.007
0.999	0.003	1.001	-0.003	0.999	0.003	1.001	-0.003
0.999	0.003	1.001	-0.003	1.001	-0.003	0.999	0.003
0.998	0.001	1.002	-0.001	1.002	-0.001	0.998	0.001
0.998	0.001	1.002	-0.001	0.998	0.001	1.002	-0.001
0.999	0.001	1.001	-0.001	0.999	0.001	1.001	-0.001
0.999	0.001	1.001	-0.001	1.001	-0.001	0.999	0.001
1.000	0.000	1.000	0.000	1.000	-0.001	1.000	0.001
0.999	0.001	1.000	-0.001	0.999	0.001	1.000	-0.001
1.000	0.001	1.000	-0.001	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000

$(Z_p^{(l)})^{-1}$ A		$(Z_p^{(-l)})^{-1}$ A		$(Z_p^{(++)})^{-1}$ A		$(Z_p^{(+-)})^{-1}$ A	
real	imag	real	imag	real	imag	real	imag
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000

$(Z_p^{(I)})^{-1}$ A		$(Z_p^{(-I)})^{-1}$ A		$(Z_p^{(++)})^{-1}$ A		$(Z_p^{(+-)})^{-1}$ A	
real	imag	real	imag	real	imag	real	imag
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000

$(Z_p^{(I)})^{-1}$ A		$(Z_p^{(-I)})^{-1}$ A		$(Z_p^{(++)})^{-1}$ A		$(Z_p^{(+-)})^{-1}$ A	
real	imag	real	imag	real	imag	real	imag
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000

$(Z_p^{(l)})^{-1} A$		$(Z_p^{(-l)})^{-1} A$		$(Z_p^{(++)})^{-1} A$		$(Z_p^{(+-)})^{-1} A$	
real	imag	real	imag	real	imag	real	imag
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000

$(Z_p^{(I)})^{-1}A$		$(Z_p^{(-I)})^{-1}A$		$(Z_p^{(++)})^{-1}A$		$(Z_p^{(+-)})^{-1}A$	
real	imag	real	imag	real	imag	real	imag
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000
Condition Number							
38.3810		45.2861		45.2861		17.4300	