National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file *Votre référence*

Our file *Notre référence*

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

Investigations of Dispersion Characteristics of Microwave/Optical Guided Wave Structures by Using Transmission Line Matrix Method

By

Bing Qiu

A thesis submitted to the
School of Graduate Studies and Research
in partial fulfillment of the requirements
for degree of

Master of Applied Science

Ottawa-Carleton Institute for Electrical Engineering
Department of Electrical Engineering
Faculty of Engineering
University of Ottawa

© Bing Qiu, Ottawa, Canada, 1993

National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-85856-7

Canada

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

I hereby declare that I am the sole author of this document. I authorize the University of Ottawa to lend this document to other individuals or institutions for the purposes of scholarly research.

Bing Qiu

I further authorize the University of Ottawa to reproduce this document by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purposes of scholarly research.

Bing Qiu

Contents

List of Figures

Acknowledgments

I would like to thank my supervisor, Dr. Michel M. Ney, for his help and guidance throughout this research project.

Special thanks also due to Mr. Z. Chen for his helpful suggestions, and to my fellow students in the microwave group for fruitful discussions.

The financial assistance of the Telecommunication Research Institute of Ontario (TRIO) and the University of Ottawa during the period of this research has been appreciated.

Finally, but by no means least, I would like to express my thank to my wife and parents for their encouragement and comprehension, which really has made this effort possible.

Abstract

This thesis presents a new three-dimensional space (3D) condensed node TLM space (Transmission Line Matrix) algorithm for the analysis of dispersion characteristics of microwave/ optical guided wave structures. This new algorithm requires much less computer memory and computation time than the conventional 3D condensed node TLM.

The application of this new algorithm to the modelling of inhomogeneous, lossy, and unbounded guided structures is investigated. More specifically, techniques such as variable mesh size, absorbing boundaries conditions, are introduced in the algorithm. The error sources, effects due to excitation and output point location are analysed and suggestions for more efficient and accurate modelling of complicated geometries are performed. In particular, special treatment is implemented to accurately model curved boundaries.

Finally, the dispersion characteristics of various kinds of rectangular and circular guided structures, are calculated by using this new algorithm. It is found that the results obtained agree fairly well with those observed by theoretical or other proved numerical methods.

Chapter 1

Introduction

1.1 Introduction

The increasing needs for different types of microwave/optical waveguides and microstrip lines have recently created considerable interest in the study of dispersion characteristics of these structures because their properties have to be known for proper design. Analytical solutions based on solving Maxwell's equations are only available for few structures and require a lot of mathematical pre-processing. With the development of digital computers, numerical methods such as FEM (Finite Element Method), MoM (Moment Method) and FDTD (Finite-Difference Time-Domain), have been developed to accurately evaluate propagation characteristics of the guided wave structures. As old and powerful numerical methods, FEM and MoM were mainly developed for solving Maxwell's equations in frequency domain. Generally, these methods need significant mathematical pre-processing before implementation in a digital computer. Taking the advantages of analyzing the electromagnetic field in time-domain technique like FDTD, which numerically seeks the direct solution to the maxwell's time-dependent curl equations, P.B.Johns [1] introduced TLM method in 1971. The principle of this method is very simple and based on the Huygens' theory of wave propagation and most compatible with today's digital computers. After

Johns' pioneer work, a lot of work [2-7]has been reported to validate this method and make it applicable for inhomogeneous, lossy, anisotropic and even nonlinear materials. It is now widely accepted as a versatile tool in the full-wave analysis of guided structures. Although this method features some advantages of analysis such as modelling nonlinear materials, it needs more CPU time and memory (particularly when utilized in analysis of frequency selective problems) than a specialized code based on extensive analytical pre-processing or other numerical methods. This makes TLM not very attractive for design purposes.

The requirement for large memory and CPU time in the analysis of frequency selective problems lie in the fact that three-dimensional meshes has to be employed. For example, to obtain the dispersion characteristics of the guide structure, one of the approaches is to resonate a section of the guided structure by placing two short-circuited planes along z axis a distance L apart. The length L corresponds to half a guided wavelength of the interested modes. The resonance frequency at the cavity corresponds to the frequency at which this particular wavelength is valid. The relationship between the propagation constant and L is then $\beta=\pi/L$. By changing L therefore β changes. One can obtain the resonance frequency of the resonator for different β. As a result, the dispersion characteristics of the guided structure can be obtained. To account for the effects of resonator along z axis, space discretization along z axis must be employed and this problem becomes a three-dimensional one and hence requires more memory. In addition, in order to maintain the modelling accuracy at higher frequency, smaller mesh size is needed. This will increase memory requirement further and may not be realized on the computer. For example, only up to 40×40×40 meshes can be discretized along x, y, z axes, respectively on Ultrix 3100 workstations when using three-dimensional condensed node TLM method. Any further increase in mesh size will exhaust computer resources.

To alleviate these problems, a new approach for TLM that uses only a two-dimensional mesh consisting of a three-dimensional space grid for the analysis of hybrid modes was proposed[28]. It is considered in this approach that, the two-dimensional case with the wave propagation direction is replaced by introducing a phase shift $\beta\Delta l$, Δl being the mesh size. This approach overcomes the shortage that only cut-off frequency of single mode in the guided structures can be obtained in two-dimensional TLM while keeping the advantages of two-dimensional analysis. Since the simulation value of each iteration along propagation direction is not changed except a phase shift, these values are not necessary to be stored in the memory for the next iteration analysis, and consequently a lot of memories can be saved. By choosing a proper propagation constant, which is valid for the selected frequency, and exciting the system with a time-domain impulse, one obtains the time-domain responses of the system after a reasonable number of iteration steps. The stored time-domain data are then processed via Fourier transform to generate frequency domain solutions. The first peak in frequency domain corresponds the frequency for which this propagation constant is valid. Higher order modes correspond to other peaks in the frequency domain spectrum. This step must then be repeated for different propagation constants to obtain the dispersion curve for one particular mode. Since only two-dimensional meshes are employed, the convergence rate is much faster than conventional 3D-TLM approach and the memory space is significantly reduced.

In this thesis, the application of this new TLM algorithm is extended to other complicated guided wave structures consisting of lossy, inhomogeneous and anisotropic materials. The dispersion characteristics of open structures, which require absorbing boundary conditions, variable mesh size during simulation in order to model open problems and the area where the field is highly variable are discussed. Finally, this method is applied to circular

guided wave structures which need special treatment of boundaries. The validation of this method applied to various guided structures is achieved by analytical solutions or other proved numerical methods.

1.2 Organization

Chapter 2 reviews the development of TLM method, from the birth of this method to the updated techniques. Some of the important literatures which reflect the development in this field are briefed. It is found from these literatures that TLM is a powerful and versatile method in the full analysis of electromagnetic wave.

In chapter 3, the basic principle of 2D and 3D expanded node TLM method is reviewed. A 3D symmetrical condensed node TLM algorithm is introduced and its applications to the modelling of lossy, inhomogeneous as well as anisotropic material, the treatment of boundaries, the excitation and output values, the sources of errors, etc., are elaborated.

Chapter 4 introduces a new 2D algorithm based on 3D symmetrical condensed node space grid to calculate the dispersion characteristics of guided wave structures.

In chapter 5, the boundaries which are placed halfway between the nodes, are reviewed. This approach is not flexible and produces some inaccuracies, particularly when curved boundaries have to be modeled. These problems can be solved by introducing reflection coefficient modification technique on boundaries that are not positioned half-way between nodes.

Chapter 6 presents and discusses the dispersion characteristics of different guided wave structures and compares them with those obtained by other methods.

1.3 Motivation

The problem of characterizing the guided wave structures in their operational modes has been a classical problem in electromagnetics for many years. Measurement of the dispersion characteristics of the guided wave structures might be expensive and impractical in many cases. With the development of digital computer, the evaluation of characteristics of these structures can be done on the computer by numerical methods such as TLM, FDTD and FEM.

Among all TLM algorithms, 3D-TLM symmetrical condensed node TLM is in particular attractive for guided wave problems because of its smaller mesh dispersion compared with other TLM algorithms. Application examples are reported in [21-29]. Although this method has many attractive features for time-domain problems, the fact that it requires large amounts of memory space and CPU time, in particular for the analysis of inhomogeneous, open and other complicated guided structures. The motivation of the work that follows was to provide an accurate way of evaluating the dispersion characteristics of guided wave structures by using a new TLM algorithm which requires less memory space and CPU time.

Very few works have been done to model the open structures by using 3D-TLM symmetrical condensed node TLM method. Among them, most of the works[25-27] were devoted to the formulation of the Absorbing Boundary Conditions (ABC). No applications of TLM algorithm to the characterization of open guided wave structures have been reported. This work has been motivated to validate the newly developed TLM algorithm by the use of the proved ABC algorithm.

Another emphasis of this work is placed on the modelling of curved boundaries be-

cause traditional staircase approximation of the curved boundary is not accurate enough, particularly when coarse mesh size is employed. A new treatment of curved boundary is proposed and implemented in the new TLM algorithm.

Chapter 2

Literature Survey

In the past two decades, frequency-domain method such as FEM and MoM have been widely used[30-39] to characterize guided wave structures. These methods, in principle, are accurate. But they need significant mathematical pre-processing. In addition, the data for the whole frequency range are calculated one frequency at a time. Therefore, it is an expensive task when the results of a wide frequency range are sought. This led us to seek an alternative way of calculating the frequency-domain solutions. Since a pulse response contain all the information of a system for the whole frequency range, it is a natural approach to use a pulse in the time-domain to excite the system, and from the time-domain pulse response to extract the frequency-domain characteristics of the system via the Fourier transform.

One numerical scheme which can be used to calculate the time-domain fields is the time-domain finite difference (FDTD) method. It was first proposed by K.S.Yee in 1966 and has been used by many investigators to solve electromagnetic problems. It is the most direct time-domain method from a mathematical point view. But it has been found[39] that the Fourier transform of the time-domain results is very sensitive to numerical errors. Therefore, even though the time domain results may be reasonably accurate, the frequency-domain results obtained from their Fourier transform may not be acceptable as useful data.

In 1971, Johns and Beurle[1] introduced a novel numerical technique for solving two-dimensional scattering problems based on the Huygens' principle of light propagation. This method, called Transmission-line Matrix (TLM) method, employs a Cartesian mesh of shunt-connected two-wire transmission-lines as a discretized propagation medium. The nodes of this mesh act as scattering centers for short voltage impulses. Thus, electromagnetic fields are modeled by filling space with a network of transmission lines. Impulses launched in the network are scattered at all nodes at a fixed time step sequence. The voltages and currents at all nodes are equivalent to electromagnetic fields in the discretized space. Later, Johns and Akhtarzad [2-7] extended this method to the modelling of inhomogeneous and lossy materials in two-dimensions. But there are only few guided wave problems related to two-dimensional case. Examples include the propagation of TE_{n0} modes in rectangular homogeneous waveguides and only the cut-off frequencies of hybrid modes can be computed in inhomogeneous guiding structures[8]. In 1974, Johns and his coworkers[9] proposed a three-dimensional expended node for the TLM method for solving the most general electromagnetic field problems in the 3D space and time. Building upon the groundwork laid by these original authors, a lot of applications have been found on the full wave analysis of guided wave problems[10-16]. In 1985, Hoefer[17] presented a very useful overview of the theory and applications of the 2D and 3D expanded node TLM methods. Every since then, a number of techniques in this field such as graded mesh technique, scalar TLM and error correction technique etc. have been developed for these algorithms. Although this 3D expanded node algorithm is by far the simplest one, it fails to describe the electromagnetic field at the same location, limiting its extensive use. To overcome these problems, P.B.Johns[18-19] developed a new 3D-TLM algorithm, using a 3D-TLM symmetrical condensed node, based on the principle of energy conservation. Allen

et al[20] tested symmetrical condensed node, expanded node and other TLM algorithms and then affirmed that the symmetrical condensed node is more accurate than any other mesh schemes. For this reason, a 3D symmetrical condensed node is used to deal with all the problems concerned in this thesis.

In addition to mesh dispersion, the symmetrical condensed node yields some significant error due to the insufficient resolution of the fields around edges and corners. This results in a 10% shift towards lower frequencies for resonator, dispersion characteristics of guide structures and filter frequency response[21]. A classical remedial approach is to use finer mesh in the vicinity of edges or corners but, unfortunately, results in large computer cost. Hoefer[22-23] proposed a method which introduces a fifth arm in 2D-TLM node and three more stubs in 3D symmetrical condensed node to account for corners and sharp edges. It was shown that the resonant frequency of a cavity containing sharp edges is accurately computed without mesh refinement around edges by using this technique. There is a net gain in terms of computer expenditure due to the fact that maximum size of the mesh dictated by the TLM can be used throughout the structure under investigation. However, for the 3D case, this technique introduces more stubs and, as a result, compensate for the gain in memory requirement as compared to mesh refinement techniques for comparable accuracy.

Another source of error is the introduction of Absorbing Boundary Conditions (ABC) to model open problems. Their algorithms can not predict accurate values of the field over a wide frequency range at the boundaries. Up to present, a number of ABC algorithms have been developed by various authors to solve open structures. Eswarappa *et al* [24] proposed a dissipation technique by which the waves striking the absorbing boundary are gradually attenuated and absorbed at boundaries. This approach needs a lot of meshes

to simulate the absorbing volume and hence a large memory requirement. Ulf *et al* [25] applied modified FDTD-ABC to 3D-TLM and yields acceptable reflection. But this is not suitable for 3D symmetrical condensed node algorithm due to spurious modes. Chen [26-27] discussed other ABC algorithms for 3D symmetrical condensed node to absorb both physical modes and spurious modes by using Taylor's and Higdon's expansion formula. Nearly zero reflection coefficient at the boundary and numerical stability were claimed. Another ABC algorithm proposed by Saguet[46], also based on Taylor's expansion formula, was applied to the 3D-TLM symmetrical condensed node and satisfactory results were reported. Taylor's ABC is used in this work because it is accurate and stable as well as easy to implement in the modified 3D-TLM algorithm for the evaluation of propagation characteristics of open guided structures.

Chapter 3

3D-TLM Symmetrical Condensed Node

3.1 Review of 2D and 3D-TLM methods

With the development of the digital computer, powerful numerical techniques have emerged to predict directly the behavior of the field quantities in time or frequency domain. As a novel time-domain method, the TLM was introduced in 1971 by Johns[1] to solve two-dimensional scattering problems. Based on a discretized version of Huygens's model of wave propagation, this two-dimensional TLM shown in Fig.3.1 consists of a mesh of shunt-connected dispersionless transmission lines. Voltage and current impulses travelling on these lines are scattered at every time-step at the nodes, simulating fields propagation. The advantage of the TLM network model is that it can be analyzed with traditional transmission-line theory. Indeed, Johns [1] has shown in this way that the voltages and currents on the TLM mesh obey the same formalism as the electric and magnetic fields of Maxwell's equations. This proves that this TLM model is consistent with Maxwell's equations and can model two-dimensional wave propagation.

In the most general case, impulses incident on all branches of any node at time $k \Delta t$ are scattered at that node, and then the reflected impulses are incident to all neighboring

nodes for the next scattering event at time $(k+1)\,\Delta\,t$.

This series of events can be described in symbolic form as follows:

$$_{k+1}[V]^r = [S]_k[V]^i;\ _{k+1}[V]^i = [C]_{k+1}[V]^r \tag{3.1}$$

where $[S]$ is the scattering matrix of the nodes, and $[C]$ is a connection matrix describing the topology of the network. The subscripts k and $k+1$ denote the discrete time intervals separating the scattering events. In this general form, the TLM algorithm applies to all two-dimensional and three-dimensional schemes.

In summary, the TLM algorithm generates the digital response of a TLM mesh to a discretized excitation. The excitation function can be a single impulse or a sequence of impulses which sample a continuous waveform in space and time. The total node voltage, for one node without stubs, at time $k\Delta t$ is computed from the branch voltages:

$$_kV_{node} = (1-\frac{1}{2})(_kV_1^i +_k V_2^i +_k V_3^i +_k V_4^i) = \frac{1}{2}\sum_{m=1}^{4} {}_kV_m^i \tag{3.2}$$

The subscripts 1 to 4 designate the four branches of a node. The time and space envelope of these samples emulates a continuous wave function. In the case of an arbitrary time dependence of the envelope, its spectral components can be obtained by discrete or fast Fourier transform of the TLM response, as stated previously. It can be shown using the duality of electric and magnetic fields that wave propagation can also be modelled by a series-connected mesh of transmission-lines [8].

Since there are only few guided wave problems related to the two-dimensional case, Akhtarzad and Johns[9] presented their so-called expanded three-dimensional node. This node is constructed by interweaving shunt and series nodes in all three coordinates. Fig.3.2 shows the unit cell of such a three-dimensional TLM network. It contains three series

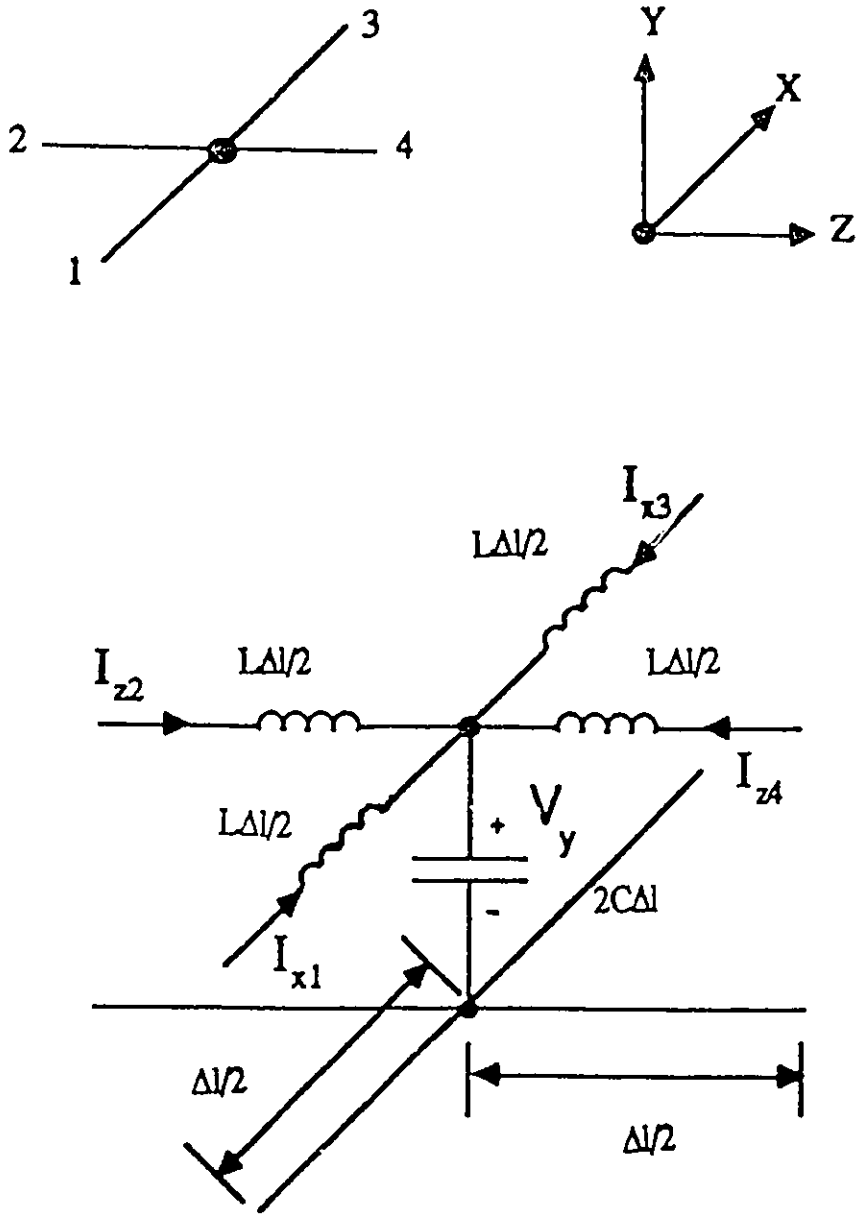

Figure 3.1: Schematic equivalent of a TLM shunt node

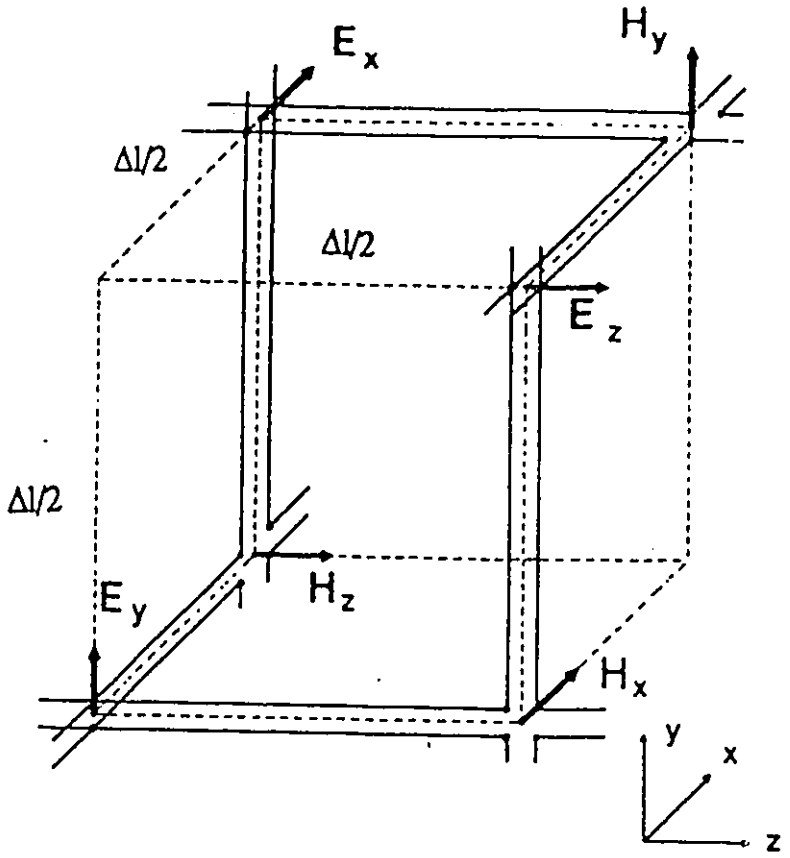

Figure 3.2: The complete three-dimensional TLM cell featuring three series and three shunt nodes

and three shunt nodes, each representing one of the six field components. The impulse scattering at these nodes has been described in the previous section.

Such a three-dimensional network is easily derived from the two dual-dimensional nodes. Unfortunately, the six field components are defined at distance $\Delta l/2$ from each other. From this reason, a number of condensed nodes were developed[8][9]. Allen *et al* [20] reported that among them, the symmetrical condensed node is the most accurate in terms of dispersion.

3.2 3D symmetrical Condensed Node

The construction of the three-dimensional symmetrical condensed node (SCN) TLM is different from that of the TLM expanded node. In stead of representing the fields at the node by voltage or current of a lumped element circuit, they are represented by voltages of transmission-line mounted square ducts made of insulating material. The symmetrical condensed node is shown in Fig.3.3. Each of the six branches represents two transmission lines carrying the two possible polarizations of the wave travelling in a given direction. These lines have the same characteristic impedance of that of free space. An impulse incident on a port is only scattered into the ports carrying the polarizations involved in Maxwell's equation. Twelve pulses on the link transmission-lines incident upon the node, produce a scattering that yield 12 reflected impulses. The voltage impulses appear on the terminals of the transmission-lines at ports which are numbered and directed as shown in Fig.3.3. The scattering matrix of the node is derived directly from Maxwell's equations and energy conservation condition. It is defined by:

$$V^r = SV^i \tag{3.3}$$

where V^i,V^r contain the incident and reflected voltages at the ports, respectively. S is the node scattering matrix which contains 12×12 elements.

Associating, for example, with voltage impulse V_1^i incident at port 1, the field components E_x and H_z, the field equation that involves coupling between these two quantities for homogeneous and sourceless material is:

$$\frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} = \varepsilon \frac{\partial E_x}{\partial t} \tag{3.4}$$

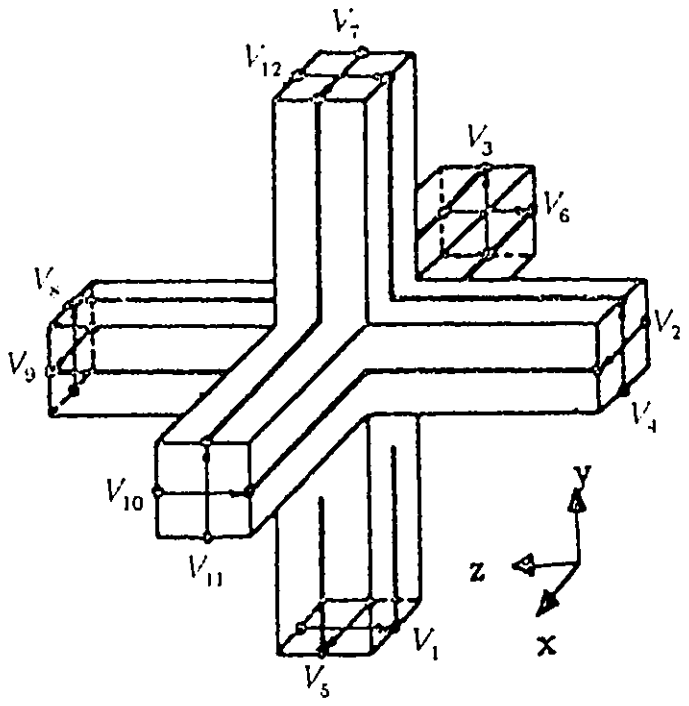

Figure 3.3: Three-dimensional TLM symmetrical condensed node

Since E_x and H_z are associated with port 12 on a y-directed line, and E_x and H_y are associated with port 2 and 9 on z-directed lines, the pulse incident on port 1 will scatter into port 1, 2, 9,and 12 after impinging the center of the node.

The amplitude of scattered pulses at port 1, 2, 9, 12 are denoted $S_{1,1}$,$S_{1,2}$,$S_{1,9}$,$S_{1,12}$ respectively. Since we are using a symmetrical condensed node in which the distance between the terminal of each port and the center of node is equal to $\Delta l/2$, $S_{1,2}$ must be equal to $S_{1,9}$, i.e, $S_{1,2}$=$S_{1,9}$=b. For simplicity, let $S_{1,1}$=a, $S_{1,12}$=c.

The other equation governing E_x and H_z is:

$$\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y} = -\mu \frac{\partial H_z}{\partial t} \tag{3.5}$$

This means that pulse excited at part 1 also scatters into port 3 and 11, and if the symmetrical node exists, there will be equal and opposite in sign. i.e.$S_{1,3}$=-$S_{1,11}$=d. Since no other components among E_y,E_z,H_x,H_y are coupled with E_x and H_z, the impulse incident

at port 1 will not scatter into ports 4, 5, 6, 7, 8, 10.i.e.

$$S_{1,4} = S_{1,5} = S_{1,6} = S_{1,7} = S_{1,8} = S_{1,10} = 0 \tag{3.6}$$

By following the same procedure at each port, each element of the S matrix can be represented by a,b,c,and d[19].

$$S = \begin{bmatrix} a & b & d & 0 & 0 & 0 & 0 & 0 & b & 0 & -d & c \\ b & a & 0 & 0 & 0 & d & 0 & 0 & c & -d & 0 & b \\ d & 0 & a & b & 0 & 0 & 0 & b & 0 & 0 & c & -d \\ 0 & 0 & b & a & d & 0 & -d & c & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & d & a & b & c & -d & 0 & b & 0 & 0 \\ 0 & d & 0 & b & a & b & 0 & 0 & -d & c & 0 & 0 \\ 0 & 0 & 0 & -d & c & b & a & d & 0 & b & 0 & 0 \\ 0 & 0 & b & c & -d & 0 & d & a & 0 & 0 & b & 0 \\ b & c & 0 & 0 & 0 & -d & 0 & 0 & a & d & 0 & b \\ 0 & -d & 0 & 0 & b & c & b & 0 & d & a & 0 & 0 \\ -d & 0 & c & b & 0 & 0 & 0 & b & 0 & 0 & a & d \\ 0 & b & -d & 0 & 0 & 0 & 0 & 0 & b & 0 & d & a \end{bmatrix} \tag{3.7}$$

A lumped equivalent circuit for this node can not be derived because the scattering at the junction is due to coupling of the fields rather than scattering at circuit nodes. Therefore, the constants in the S matrix can not be represented by values of circuit lumped elements. But, fortunately, they can be derived by enforcing the principle of energy conservation. For loss-free propagation wave, this leads to the requirement that [S] be a unitary matrix[36], i.e.,

$$S^T S = I \tag{3.8}$$

which results in the following equations[19]:

$$a^2 + 2b^2 + c^2 + 2d^2 = 1 \tag{3.9}$$

$$2ab + 2bc = 0 \tag{3.10}$$

$$2ad - 2cd = 0 \tag{3.11}$$

$$2ac + 2b^2 - 2d^2 = 0 \tag{3.12}$$

By solving these four equations to yield:

$$a = 0, c = 0, b = 1/2, d = 1/2 \tag{3.13}$$

Now the scattering matrix represented by a,b,c,d becomes:

$$S = \frac{1}{2}\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \tag{3.14}$$

Reflected voltage at the node can be now expressed as the linear combination of incident voltages. For example, at time $k\Delta t$, the reflected voltage at port 1 of a node (i, j, k) in space is the summation of the incident voltages at port 2,3,9,11:

$${}_kV_1^r(i,j,k) = 1/2[{}_kV_2^i(i,j,k) +_k V_3^i(i,j,k) +_k V_9^i(i,j,k) -_k V_{11}^i(i,j,k)] \tag{3.15}$$

The reflected voltages at time $k\Delta t$ will become the incident voltages for next time step of corresponding adjacent ports:

$${}_{k+1}V_{12}^i(i,j-1,k) = {}_kV_1^r(i,j,k) \tag{3.16}$$

$${}_{k+1}V_9^i(i,j,k-1) = {}_kV_2^r(i,j,k) \tag{3.17}$$

$${}_{k+1}V_{11}^i(i-1,j,k) = {}_kV_3^r(i,j,k) \tag{3.18}$$

$${}_{k+1}V_8^i(i,j,k-1) = {}_kV_4^r(i,j,k) \tag{3.19}$$

$${}_{k+1}V_7^i(i,j-1,k) = {}_kV_5^r(i,j,k) \tag{3.20}$$

$${}_{k+1}V_{10}^i(i-1,j,k) = {}_kV_6^r(i,j,k) \tag{3.21}$$

$$_{k+1}V_5^i(i,j+1,k) = {}_kV_7^r(i,j,k) \quad (3.22)$$

$$_{k+1}V_4^i(i,j+1,k) = {}_kV_8^r(i,j,k) \quad (3.23)$$

$$_{k+1}V_2^i(i,j,k+1) = {}_kV_9^r(i,j,k) \quad (3.24)$$

$$_{k+1}V_6^i(i+1,j,k) = {}_kV_{10}^r(i,j,k) \quad (3.25)$$

$$_{k+1}V_3^i(i+1,j,k) = {}_kV_{11}^r(i,j,k) \quad (3.26)$$

$$_{k+1}V_1^i(i,j,k+1) = {}_kV_{12}^r(i,j,k) \quad (3.27)$$

Thus, it is only necessary to store the incident voltages for each node and each of their 12 branches at every iteration. Note that the TLM method is completely iterative, and requires no matrix inversion, thus making it very stable.

3.3 Wave Properties of 3D Symmetrical Condensed Node

Unlike continuous wave models which are valid at all frequencies, discretized models such as TLM, simulate a continuous systems as long as the mesh parameter Δl is much smaller than the wavelength of operation. The TLM mesh being a periodic structure, its wave properties exhibit alternating pass-band and stop-band characteristics. In other words, the discrete model has spurious solution at wavelengths which are of the order of the mesh parameter Δl. It is important to study these wave properties in order to understand the limitations of the TLM modelling procedure, eventually to determine and to correct errors due to the finite length of mesh parameter Δl.

The dispersion relation of the 3D condensed node was first studied in the special case of as propagation along the three coordinate axes and along the diagonals. Nielson[21] derived the eigenvalue equation of dispersion characteristics for the 3D symmetrical condensed node

by using Floquent's theorem. A closed form expression of the simulated wave propagation vector was derived for arbitrary propagation. The dispersion characteristics of the 3D symmetrical condensed node were compared with the expanded node. Superior dispersion performances were reported for the symmetrical condensed node. However, the symmetric condensed node is subjected to spurious mode generation yielding to some instabilities in certain case[27].

3.4 Modelling of Lossless Inhomogeneous Media

The scattering matrix proposed in previous section has been used to solve scattering problems which are assumed to be filled with homogeneous medium. However, there is a large class of inhomogeneous guided wave structures in which dispersion characteristics has to be determined.

Extra stubs are introduced, like for 2D-TLM, to model constitutive parameters. For the SCN, six extra ports have to be employed. Stubs donated 13,14,15,16,17,18 models six field components E_x,E_y,E_z,H_x,H_y,and H_z respectively. Although the node can not be represented by an equivalent lumped element network, the concept that electric field can be modeled by the voltage across the capacitor of the node still holds. To achieve this, E-field stub must be open-circuited and thus add some capacitance to the node. Similarly, H-field stub must be short-circuited and add some inductances to the node. The length of these additional ports is also $\Delta l/2$ in order to maintain the synchronism. The time taken for a pulse to travel from the port terminals to the center of the node where scattering events take place is $\Delta t/2$, where Δt is the TLM time step.

To determine the scattering matrix of the node loaded with stubs, a similar analysis can be made for all arms. Since, by definition, the voltage pulse incident on port 13 only

couples with the E_x field, it will scatter into ports 1,2,9,and 12. Because of symmetry of the node, the scattered amplitudes of these ports excited by an impulse at port 13 are equal and assigned to e, while the amplitude scattered back into itself is assigned to be h. Also pulses incident on port 1,2,9, and 12 will couple into port 13 and, for the same reason are equal and assigned to g. Similar analysis applied to other ports results in the following scattering matrix[19]:

$$\begin{bmatrix}
a & b & d & 0 & 0 & 0 & 0 & 0 & b & 0 & -d & c & g & 0 & 0 & 0 & 0 & i \\
b & a & 0 & 0 & 0 & d & 0 & 0 & c & -d & 0 & b & g & 0 & 0 & 0 & -i & 0 \\
d & 0 & a & b & 0 & d & 0 & b & 0 & 0 & c & -d & 0 & g & 0 & 0 & 0 & -i \\
0 & 0 & b & a & d & 0 & -d & c & 0 & 0 & b & 0 & 0 & g & 0 & i & 0 & 0 \\
0 & 0 & 0 & d & a & b & c & -d & 0 & b & 0 & 0 & 0 & 0 & g & -i & 0 & 0 \\
0 & d & 0 & 0 & b & a & b & 0 & -d & c & 0 & 0 & 0 & 0 & g & 0 & i & 0 \\
0 & 0 & 0 & -d & c & b & a & d & 0 & b & 0 & 0 & 0 & 0 & g & i & 0 & 0 \\
0 & 0 & b & c & -d & 0 & d & a & 0 & 0 & b & 0 & 0 & g & 0 & -i & 0 & 0 \\
b & c & 0 & 0 & 0 & -d & 0 & 0 & a & d & 0 & b & g & 0 & 0 & 0 & i & 0 \\
0 & -d & 0 & 0 & b & c & b & 0 & d & a & 0 & 0 & 0 & 0 & g & 0 & -i & 0 \\
-d & 0 & c & b & 0 & 0 & 0 & b & 0 & b & a & d & 0 & g & 0 & 0 & 0 & i \\
c & b & -d & 0 & 0 & 0 & 0 & 0 & b & 0 & d & a & g & 0 & 0 & 0 & 0 & -i \\
e & e & 0 & 0 & 0 & 0 & 0 & 0 & e & 0 & 0 & e & h & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & e & e & 0 & 0 & 0 & e & 0 & 0 & e & 0 & 0 & h & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & e & e & e & 0 & 0 & e & 0 & 0 & 0 & 0 & h & 0 & 0 & 0 \\
0 & 0 & 0 & f & -f & 0 & f & 0 & -f & 0 & 0 & 0 & 0 & 0 & 0 & j & 0 & 0 \\
0 & -f & 0 & 0 & 0 & f & f & 0 & f & -f & 0 & 0 & 0 & 0 & 0 & 0 & j & 0 \\
f & 0 & -f & 0 & 0 & 0 & 0 & 0 & 0 & 0 & f & -f & 0 & 0 & 0 & 0 & 0 & j
\end{bmatrix} \tag{3.28}$$

This is a 18×18 matrix in which the value of each element is associated with the characteristic admittance and impedance of corresponding port. By applying current and voltage conservation as well as unitary condition as before, the values of the elements in the scattering matrix are expressed as[19]:

$$a = \frac{-Y}{2(4+Y)} + \frac{Z}{2(4+Z)} \tag{3.29}$$

$$b = \frac{4}{2(4+Y)} \tag{3.30}$$

$$c = \frac{-Y}{2(4+Y)} - \frac{Z}{2(4+Z)} \tag{3.31}$$

$$d = \frac{4}{2(4+Z)} \tag{3.32}$$

$$e = b \tag{3.33}$$

$$f = Zd \tag{3.34}$$

$$g = Yb \tag{3.35}$$

$$h = \frac{Y-4}{Y+4} \tag{3.36}$$

$$i = d \tag{3.37}$$

$$j = \frac{4-Z}{4+Z} \tag{3.38}$$

where Y and Z take a subscript appropriate to the corresponding stub, for example,

$$S_{2,9} = c = \frac{-Y_x}{2(4+Y_x)} - \frac{Z_y}{2(4+Z_y)} \tag{3.39}$$

Other matrix elements value appropriate to the corresponding stub can be found in [19]. The values of the inductive (Z) and capacitive (Y) stubs are given by:

$$Y_x = 4(\epsilon_{xx} - 1) \tag{3.40}$$

$$Y_y = 4(\epsilon_{yy} - 1) \tag{3.41}$$

$$Y_z = 4(\epsilon_{zz} - 1) \tag{3.42}$$

$$Z_x = 4(\mu_{xx} - 1) \tag{3.43}$$

$$Z_y = 4(\mu_{yy} - 1) \tag{3.44}$$

$$Z_z = 4(\mu_{zz} - 1) \tag{3.45}$$

$\epsilon_{xx}, \epsilon_{yy}, \epsilon_{zz}$ and $\mu_{xx}, \mu_{yy}, \mu_{zz}$ are the permittivity and permeability along x, y, z directions, respectively. It can be noticed that TLM method can also model non-isotropic

material provided that the principal axes coincide with the TLM mesh coordinate system. In other words, the permeability and permittivity tensor with respect to that coordinate system has to be diagonal.

3.5 Modelling of Lossy Media

In the symmetrical condensed node without losses, six transmission line stubs with a length equal to one half of the cell size are added to the node to model local permittivity or permeability. These lines are terminated into either an open or short circuit so that all the energy that couples into one of these stubs is returned to the node. In order to model losses, six more stubs have to be added to the node; three to model dielectric losses and three to model magnetic losses. These additional stubs are considered to be infinitely long (or equivalently terminated by a match load) so that no part of the pulses that couple into these stubs is returned to the node. Since pulses that couple into these stubs are not returned to the node, there is no need to store them in the computer. Therefore, the lossy node scattering matrix requires the same computer storage space as the node without losses.

To model dielectric losses, we add stubs with normalized (with respect to link lines) characteristic admittances G_{ox},G_{oy} and G_{oz} which couple only to the E_x,E_y and E_z electric field components, respectively. For magnetic losses, we add stubs with normalized characteristic impedances R_{ox},R_{oy} and R_{oz} which couple only to the H_x,H_y and H_z magnetic field components, respectively. The scattering matrix of the node is derived by forcing charge conservation to be enforced at the node and requiring the scattering matrix to be unitary; thus, the complete matrix is obtained following the same procedure as mentioned in the previous section. But the entries, a through j, are calculated differently. The new

expressions for the scattering matrix elements fully equipped with all twelve stubs are given by the following expressions, provided that the medium is isotropic in terms of losses:

$$a = -\frac{Y+G}{2(4+Y+G)} + \frac{Z+R}{2(4+Z+R)} \tag{3.46}$$

$$b = \frac{4}{2(4+Y+G)} \tag{3.47}$$

$$c = -\frac{Y+G}{2(4+Y+G)} - \frac{Z+R}{2(4+Z+R)} \tag{3.48}$$

$$d = \frac{4}{2(4+Z+R)} \tag{3.49}$$

$$e = b \tag{3.50}$$

$$f = Zd \tag{3.51}$$

$$g = Yb \tag{3.52}$$

$$h = \frac{Y-G-4}{Y+G+4} \tag{3.53}$$

$$i = d \tag{3.54}$$

$$j = \frac{4+R-Z}{4+R+Z} \tag{3.55}$$

Expressions for Y and Z and are given in previous section. The parameters G and R are calculated, for a regular Cartesian mesh, by

$$G = \sigma_e \Delta l Z_0 \tag{3.56}$$

$$R = \sigma_m \Delta l / Z_0 \tag{3.57}$$

where Δl is the TLM cell size, Z_0 is the characteristic impedance of the link lines (usually taken to be 377Ω), and σ_e are σ_m are the equivalent electric and magnetic conductivities, respectively, in the Cartesian coordinate direction.

3.6 Modelling of Boundary

The finite memory of computers limits the size of the TLM networks that can be modeled. Thus, it is necessary to limit the computational domain by introducing boundaries. Real structures such as waveguides, circuit components must be defined by boundaries. Hence, the realistic modelling of all types of boundaries is essential.

Boundaries can be classified as non-dispersive or dispersive boundaries. Non-dispersive boundaries are locations where the relationship between field quantities or reflection coefficient are frequency independent. Thus, they can be modeled by a real (time independent) reflection coefficient that relates incident and reflected impulses at their location. Electric and magnetic walls for instance, are represented by short and open circuits connected at their position in the TLM mesh, respectively. During the computation, reflected impulses from a magnetic or electric wall is computed by multiplying the incident voltage amplitude by the appropriate reflection coefficient.

Dispersive boundaries should be modeled by frequency-independent reflection coefficient, which is inconsistent with the TLM scheme in time-domain. However, for a narrow-band excitation, the reflection coefficient can be assumed constant. In addition, in the case of non-perfect conducting walls, the reflection coefficient can be computed from the surface impedance $Z_s = (1+j)R_s$ and is complex. But for high conductivities the imaginary part is negligible and is computed by:

$$\Gamma_i = (Z_s - Z_0)/(Z_s + Z_0) \simeq -1 + \sqrt{\frac{\mu\omega\epsilon_0}{\sigma\mu_0}} \tag{3.58}$$

and in the three-dimensional case:

$$\Gamma_i \simeq -1 + 2\sqrt{\frac{j\omega\epsilon_0}{2\sigma/\mu_0}} \tag{3.59}$$

Note that since Γ_i depends slightly on the angular frequency ω, the loss calculations are accurate only for that frequency which has been selected in determining Γ_i. In the case of good conductors, this approach is fully adequate over an octave bandwidth.

For open problems, artificial absorbing boundaries must be introduced in order to limit the size of the computational domain. For normal incidence of a TEM wave, an absorbing boundary can be achieved without dispersion by terminating the mesh lines by the intrinsic impedance of the medium simulated by the TLM network. However, when the angle of incidence is not normal absorbing boundary condition algorithm must be applied.

3.7 Absorbing Boundary Conditions

The difficulty encountered when trying to solve open field problems is limiting the domain of computation such that no reflection occurs where the mesh is terminated. This condition should hold over a wide frequency range and for any type of waves. One refers to Absorbing Boundary Conditions(ABC).

One of the commonly used approaches is to use extrapolation techniques, where the field values at absorbing boundary location are predicted, at each time step, from field values at interior nodes.

Quite a few absorbing boundary conditions have been developed. But they are just suitable for 3D expanded node algorithm. Saguet[39] proposed an absorbing boundary algorithm which is easy to implement and is applicable to the 3D-TLM SCN. This algorithm eliminates the effects of spurious mode as reported by Chen[27]. The principle behind

Saguet's algorithm is based on Taylor's expansion and to predict the pulses reflected by the boundary at each time step, as a function of those reflected by the interior nodes (see Fig.3.4):

For 1 node(1st order):

$$_{k+1}V^i(i,j) =_k V^r(i-1,j) \tag{3.60}$$

For 2 nodes(2nd order) :

$$_{k+1}V^i(i,j) = 2_kV^r(i-1,j) -_{k-1} V^r(i-2,j) \tag{3.61}$$

For 3 nodes(3rd order):

$$_{k+1}V^i(i,j) = 2.5_kV^r(i-1,j) + 2.0_{k-1}V^r(i-2,j) + 0.5_{k-2}V^r(i-3,j) \tag{3.62}$$

where V^r and V^i correspond to voltages of arms connecting to the boundaries. The absorbing conditions are modeled by implementing the above algorithms. The accuracy of the algorithm depends on the number of nodes used for the pulse prediction. Third order algorithm is used to simulate all problems in this work and found to be accurate when utilized in the calculation of dispersion characteristics of open guiding structures as demonstrated in chapter 5.

3.8 Excitation and Output

Launching excitation and extracting the response after a reasonable number of time steps are very important. For instance, depending on the location of the excitation, modes may not be excited or not appear in the response. Generally speaking, excitation and output points should be placed at locations where the amplitude is relatively large enough compared with the peak values.

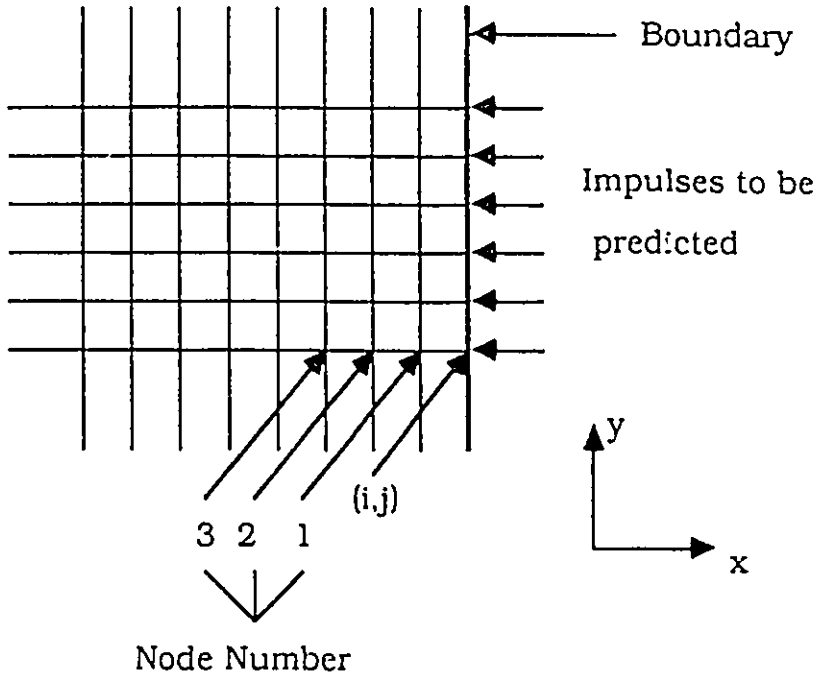

Figure 3.4: Saguet's absorbing boundary

3.8.1 Excitation

In the case of zero initial condition, the amplitudes of impulses throughout the mesh is set to zero. The network is then excited at selected source or inputs points with a Dirac function or a stream of impulses weighted by any time-function. As time progresses, impulses travel from one node to neighboring nodes to finally reach all nodes of the domain of computation. Each iteration in the computer represents a time interval dictated by the ratio $\Delta l/c$, where c is the speed of light in vacuum Since a Dirac impulse contains all frequency components, the corresponding response will also contain information over all frequencies. However, frequency dispersion of the TLM mesh will limit the range of frequency over which the response is accurate.

P.B.Johns[19] showed that in the case of the 3D-TLM SCN, the field components are

excited according to a set of relations. For instance, for a given set E_y,H_x,H_z. the voltage amplitude at the ports are:

$$V_3^i = (vE_y - wZ_0H_z)/2 \tag{3.63}$$

$$V_4^i = (vE_y + uZ_0H_x)/2 \tag{3.64}$$

$$V_8^i = (vE_y - uZ_0H_x)/2 \tag{3.65}$$

$$V_{11}^i = (vE_y + wZ_0H_z)/2 \tag{3.66}$$

$$V_{14}^i = vE_y/2 \tag{3.67}$$

$u = v = w = 1$ for a square cell. Z_0 is the free space wave impedance. For example, if only the field E_y is to be excited at a node with permittivity stub (port 14, not shown in Fig.3.3), the following incident impulse are required:

$$V_3^i = v/2, V_4^i = v/2, V_8^i = v/2, V_{11}^i = v/2, V_{14}^i = v/2 \tag{3.68}$$

3.8.2 Output

By enforcing the principle of current(charge) conservation, Johns[19] showed that E_y , for example, can be computed by:

$$E_y = 2(v_3^i + v_4^i + v_8^i + v_{11}^i + Y_y v_{14}^i)/v(4 + Y_y) \tag{3.69}$$

where Y_y was defined before. Other field components can be found in a similar manner[19]. The frequency domain solution of E_y can then be obtained by performing Fourier transform on the time-response which consists of a stream of impulses computed by (3.69).

3.8.3 Fourier Transform

As explained before, The TLM method has the advantage to provide the frequency response of a structure from the impulsive response. The output impulse function at a particular

point in the mesh is simply obtained by observing the stream of impulses and can be written as:

$$F(t) = \sum_{k=1}^{\infty} A_k \delta(t - kt_0) \tag{3.70}$$

where A_k is field amplitude at kth time step. The above time-domain function contains information within the frequency range of the TLM network. The response to any arbitrary excitation time-function can be computed from the impulse response $F(t)$ by convolution.

Of particular interest is the frequency response of time-domain solutions. Since $F(t)$ is a series of delta functions, the fourier integral is computed by summations. The real and imaginary parts of the output spectrum are:

$$Re[F(\frac{\Delta l}{\lambda})] = \sum_{k=1}^{n} A_k cos(2\pi k \frac{\Delta l}{\lambda}) \tag{3.71}$$

$$Im[F(\frac{\Delta l}{\lambda})] = \sum_{k=1}^{n} A_k sin(2\pi k \frac{\Delta l}{\lambda}) \tag{3.72}$$

Where $F(\frac{\Delta l}{\lambda})$ is the frequency response. In this application, different values of propagation constant β will correspond to the frequency for which the modulus $\mid F \mid = ([ReF]^2 + [ImF]^2)^{1/2}$ has a peak for a particular mode.

3.9 Sources of Errors

Several sources of errors including truncation error,velocity error and coarseness error have been reported[8]. All play a role in limiting the accuracy of the TLM method and it must be applied with caution in order to yield reliable and accurate results.

The need to truncate the output impulse function leads to the so-called truncation error. It is shown[8] that the truncation error decreases with increasing number of iterations, but

at the price of more CPU time. It is recommended that the number of iterations be chosen such that the truncation error is reduced to a fraction of a percent and can be neglected.

Due to the periodic structures of the TLM network, the wave propagation in the network experiences dispersion effect. These effects usually cause network simulations to deviate from actual space wave propagation, resulting in velocity error. This error responds to the same remedial measures as the coarseness error which will be discussed later, and it therefore does not need to be corrected separately.

The coarseness error occurs when the TLM mesh is too coarse to approximate highly nonuniform field which can exist at corners and edges. This error is particularly cumbersome when analysing planar structures which contain singular regions. A possible measure would be to choose a very fine mesh. However, this would lead to large memory requirements particularly for three-dimensional problem. A better approach is to introduce a network of variable mesh size to provide higher resolution in the highly varying field region. Recently, Hoefer[22] proposed a new method which employ a fifth arm attached to corners or edges. These measures are reported to reduce the coarseness error by one order of magnitude and, simultaneously, reduce the velocity error.

Chapter 4

An Improved TLM Node for Full-wave Analysis

4.1 Introduction

The 3D-TLM SCN can be also used to analyse the dispersion characteristics of guided wave structures like 3D expanded node [8]. Dispersion analysis by these TLM techniques involve resonating a section of the guided structures by placing short-circuiting planes along the axis of propagation. The separation of the shorting planes equals half of the guided wavelength for the fundamental mode at the resonant frequency of the cavity. If the distance between these planes is $2L$, the phase constant is given by $\beta=\pi/2L$. By changing L also β changes. Repeating the calculation for the resonance frequency corresponding to each new β, the dispersion characteristics of the guided structure can be then obtained.

Chio [49] *et al* applied this technique to calculate the dispersion characteristics of a simple rectangular cavity as shown in Fig.4.1. They have assumed a dominant mode in the initial value assignment. Only up to 20 meshes can be discretized along propagation direction without exhausting computer resourses. It takes about twenty minutes to obtain the resonant frequency under the conditions of 2000 iterations and maximum mesh-size discretization. The simulation results are shifted towards lower frequencies when compared

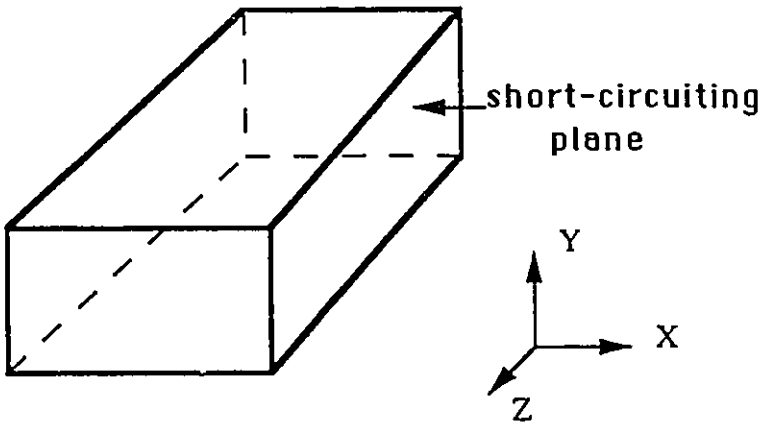

Figure 4.1: Rectangular Cavity

with theoretical solutions. This is mainly produced from the fact that the mesh size along the z-axis is too large.

The large memory space and CPU time requirement are mainly due to the fact that the full-wave analysis requires a three-dimensional mesh and that processing a time-domain impulse involves from the start much more frequency information than what is actually needed for the guided structure analysis. Only after the impulse response is obtained, a Fourier transform selects the information of interest.

In the following section, a two-dimensional TLM algorithm for full-wave analysis is presented. Based on the 3D-TLM as described in chapter 3, it reduces the memory and CPU time significantly while keeping a comparable simulation accuracy.

4.2 A new TLM node for full-wave analysis

The 3D-TLM algorithm was introduced in chapter 3. The proposed TLM full wave analysis is based on that algorithm, in which the following relationship between the incident and

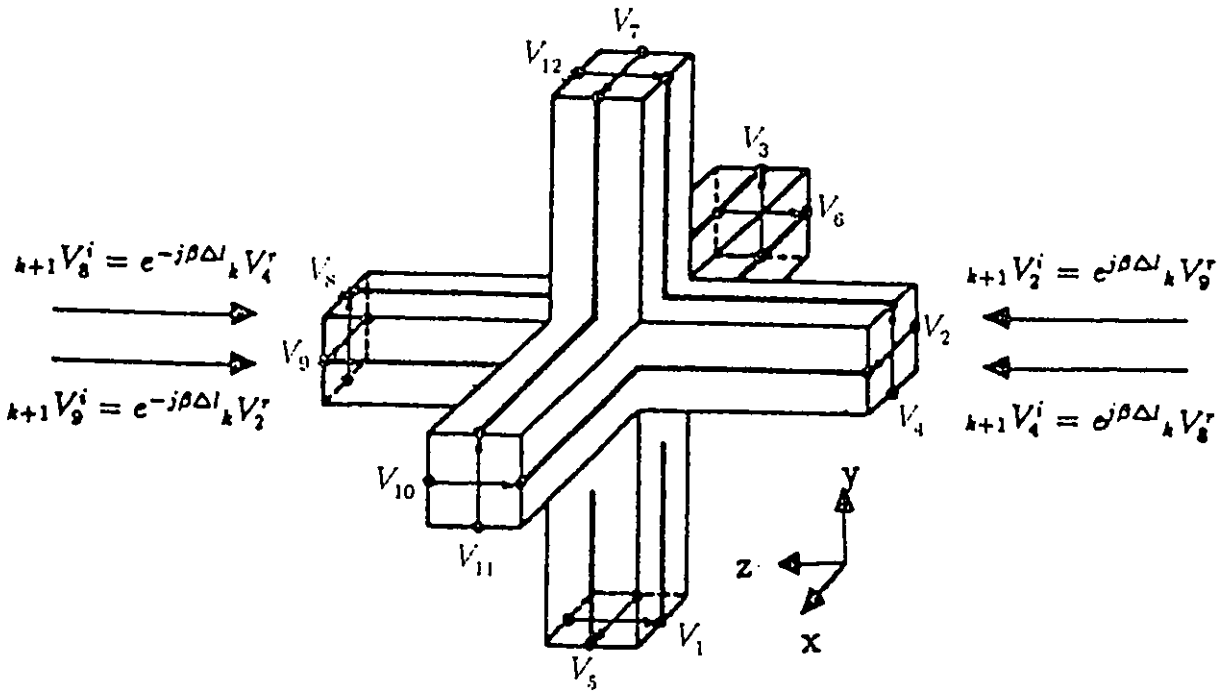

Figure 4.2: A modified 3D-TLM symmetrical condensed Node

reflected voltages at arms along the z-direction are as follows:

$$_{k+1}V_8^i(i,j,k-1) = exp(-j\beta\,\Delta\,l)_kV_4^r(i,j,k) \quad (4.1)$$

$$_{k+1}V_9^i(i,j,k-1) = exp(-j\beta\,\Delta\,l)_kV_2^r(i,j,k) \quad (4.2)$$

$$_{k+1}V_2^i(i,j,k+1) = exp(j\beta\,\Delta\,l)_kV_9^r(i,j,k) \quad (4.3)$$

$$_{k+1}V_4^i(i,j,k+1) = exp(j\beta\,\Delta\,l)_kV_8^r(i,j,k) \quad (4.4)$$

Where Δl is the mesh size. The new concept of this modified 3D symmetrical condensed node(MSCN) is shown in Fig.4.2. $_kV_4^r$,$_kV_2^r$,$_kV_9^r$,$_kV_8^r$ and $_{k+1}V_4^i$,$_{k+1}V_2^i$,$_{k+1}V_9^i$,$_{k+1}V_8^i$ are the reflected voltages at time $k\,\Delta\,t$ and the incident voltages at time $(k+1)\,\Delta\,t$ on the lines 4, 2, 9, 8, respectively. $\Delta t = \Delta l/c$, with Δl is the mesh parameter and c the velocity of light. The derivation of equations(4.1-4.4) is obvious recalling that the z-axis dependent factor of the guided mode is $exp(j\beta\Delta l)$, where β is the phase constant. Thus, if a mode has been excited, the reflected or incident voltages along the z-axis at two adjacent nodes

have only a phase difference of $\beta\Delta l$ at the resonant frequency. The incident voltages on the lines 2 and 4 at time $(k+1)\,\Delta\,t$ are the reflected voltages of the next node along the negative z-axis direction at the time $k\,\Delta\,t$, which in turn are identical to the reflected voltages on the lines 9 and 8, except there is an increase of βl in the phase. Similarly, the incident voltages on the lines 9 and 8 at time $(k+1)\,\Delta\,t$ are the reflected voltages of the next node along the positive z-axis direction at the time $k\,\Delta\,t$, which are also identical to the reflected voltages on lines 2 and 4, except there is an decrease of βl in the phase. In other words, the z-axis arms of the three-dimensional node are directly connected by a non-reciprocal phase shift.The reflected voltages on one z-axis arm at time $k\,\Delta\,t$ will propagate along this phase shift and then become the incident voltages on the other arm at time $(k+1)\,\Delta\,t$. Therefore the three-dimensional node is closed in the z-axis direction and the iteration procedure for the impulse propagation in space need only be carried out in x and y direction. This equivalently reduces the three-dimensional line mesh to only a two-dimensional network. Since β takes part into calculation as an input data, the whole dispersive characteristics can be obtained without changing the layout of the mesh for each new β value. Hence the CPU time and memory space required are reduced considerably.

It has to be pointed out that the amplitudes of impulses as shown in equation (4.1-4.4) are complex quantities and not real values as required by the conventional TLM method. It means that an impulse with a complex amplitute imposes a constant phase shift on all frequencies involved. Since the scattering and transmission matrix are real, there is no coupling between the real and imaginary parts during scattering and propagation among the nodes. When performing Fourier transform, the complex notations in the time domain are regarded as real quantities. Coupling between them occurs only through equations(4.1-4.4). Hence, developing these equations with its real and imaginary part will result in eight

real equations instead of four complex ones. By linearly combining two real solutions one obtains formally the same results as if one would use complex notation right from the beginning. The complex notation used in this thesis does not change the TLM algorithm but simplifies the formulation as well as the programming significantly.

4.3 Calculation Procedures

The model described above can be considered a transmission line resonator with length Δl in z direction. But it is different from the resonator employed in the conventional TLM method from the fact that the propagation constant β in this resonator is not related to the length Δl by $\beta = 2\pi/L$. This characteristic allows the computation for different values of β with a fixed value of Δl. Consequently, the number of the nodes in z direction is reduced to 1, which is equivalent to a two-dimensional mesh. Another novel characteristic of this resonator is that it supports travelling waves rather than standing waves (two travelling waves with same β propagating in opposite directions).

Nevertheless, the calculation steps in this new approach are similar to those in the conventional TLM full-wave analysis. First, we excite this "travelling wave resonator" by launching some initial impulses which will consequently propagate and be scattered. After a sufficient number of iterations, the field impulses response contains contribution from all modes propagating in the guided structure. Performing the Fourier transformation on the time-response determines the position of peaks in the spectrum, which corresponds to resonant frequencies of the "travelling wave resonator" for a given β. Therefore , we can build up the dispersion characteristic relating constant β and the frequency for all modes of the structure.

Chapter 5

Curved Boundary Treatment

5.1 Introduction

The transmission line matrix method for the numerical analysis of wave propagation uses a mesh of transmission lines to represent space either in two or three dimensions as presented in previous chapters. Most of the guided wave structures that have been analysed by the method have boundaries that coincide with a cartesian mesh and, therefore, can easily run between two adjacent nodes in order to satisfy synchronism. This type of modelling will result in inaccurate solutions when applied to curved boundaries since it can only represent them in a stepwise manner. Similar considerations apply when modelling of waveguide components, discontinuities or junctions requires an accurate positioning of boundaries which can not be an exact multiple of $\Delta l/2$ without using small mesh sizes. It is, therefore, important to provide an algorithm of correction for arbitrary positioning of boundaries.

This chapter introduces the techniques to treat the curved boundaries both in two and three dimensions. It is found that the new technique removes the restriction that dimensions of simulated structures can only be integer multiples of the mesh parameter and thus considerably improves the flexibility and accuracy of TLM modelling of

microwave/millimiterwave/optical components by freeing the model from the traditional staircase approximation of curved boundaries and by improving the geometrical resolution without significant increase in computational expenditure.

5.2 Piecewise Straight Boundary

Traditionally, boundaries are placed halfway between nodes in order to keep the synchronism. This condition is obvious when applied to straight-line boundary as discussed in previous chapters. This technique can be also applied to curved boundaries and leading to what is called piecewise(staircase) approximation as shown in Fig.5.1.

The implementation of TLM algorithm for curved boundaries is the same as that for staircase approximation except that the incident impulses of the nodes adjacent to the boundary(boundary nodes) have to be modified along two directions in-stead of one direction in the case of straight-line boundaries. All the boundary nodes require independent treatment since they can not be modeled in the same way as interior nodes. This is a tedious work, particularly when very fine meshes are used and a lot of boundary nodes have to be treated. As stated before, the simulation results found by using staircase approximation technique is not accurate since it does not represent the geometry of the boundary exactly. To increase the simulation accuracy, reflection coefficient modification technique in 3D condensed node is introduced in the next two sections.

5.3 Arbitrary Position of Boundary

When boundary nodes do not lie half-way between nodes, the time taken for a impulse to leave an adjacent node, bounce on the boundary, and return to the node will not be $\Delta l/c$, where Δl is the spacing between interior nodes. This is in conflict with the basic

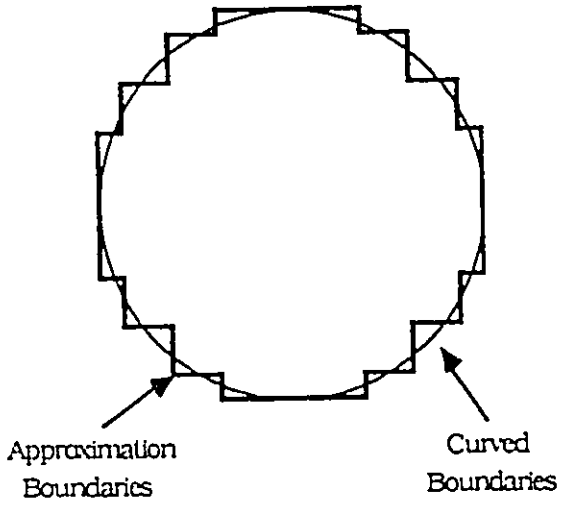

Figure 5.1: Approximation of a curved boundary by a piecewise straight boundary

principle of the TLM analysis since it requires that the impulses everywhere in network arrive at all nodes simultaneously. However, this difficulty can be overcome by stipulating that branch of transmission-line joining the node to the boundary have the same length as other branches, but that its characteristic admittance be altered to account for the difference between its true length and $\Delta l/2$.

Fig.5.2 shows the concept of arbitrary wall positioning in two-dimensional TLM. Consider a short transmission line AB of length $l = \alpha \Delta l/2$ and of unitary characteristic admittance as shown in Fig.5.2. If B is a short-circuited termination (electric wall), then the input admittance y_A at A is given by:

$$y_A = \frac{1}{jtan(\beta\alpha \Delta l/2)} \tag{5.1}$$

Where β is the phase constant for the line. A'B' is another line of length $\Delta l/2$ with characteristic admittance y, on which waves propagate with the same velocity as on AB.

If B′ is also short-circuited, then the input characteristic admittance $y_{A'}$ at A′ is:

$$y_{A'} = \frac{y}{jtan(\beta \Delta l/2)} \tag{5.2}$$

The input admittance at A and A' should be the same in order to deal with the same problems, i.e., $y_A = y_{A'}$, then, for small $\beta \Delta l$ (small $\Delta l/\lambda$):

$$y = \frac{1}{\alpha} \tag{5.3}$$

Similarly, if B and B′ are open-circuited:

$$y = \alpha \tag{5.4}$$

Since the characteristic admittance of one branch is not the same as the other three branches. The scattering matrix will be different from what is described in equation (3.2). A modification of the impulse scattering matrix of boundary node is needed. On the transmission-line matrix, if V_n^i is the magnitude of an incident pulse on a node from the nth co-ordinate direction, then the pulse V_n^r reflected into the nth coordinate direction is given by[14]:

$$V_n^r = (\sum_{m=1}^{4} T_m V_m^i) - V_n^i \tag{5.5}$$

Where:

$$T_n = \frac{2y_n}{\sum_{m=1}^{4} y_m} \tag{5.6}$$

Where y_n is the normalized admittance of the line in the nth coordinate direction. Recognizing that the reflected pulses from one node become incident pulses on a neighboring node, an iteration process can be set up from the above equations.

The curved boundary may lie at any place between two nodes as shown in Fig.5.3. We can apply the above technique along two, three or even four directions provided that we

Table One

Mode	Theoretical result	Numerical result Piecewise Boundary	Numerical result Arbitrary Boundary
E_{01}	0.3828	0.3818	0.3815
E_{02}	0.8785	0.8653	0.8679
E_{11}	0.6099	0.6020	0.6077
E_{12}	1.1166	1.0976	1.1099
E_{21}	0.8174	0.8219	0.8159
E_{22}	1.3396	1.3219	1.3280
H_{01}	0.6099	0.5978	0.6024
H_{02}	1.1166	1.0611	1.0887
H_{11}	0.2930	0.2816	0.2834

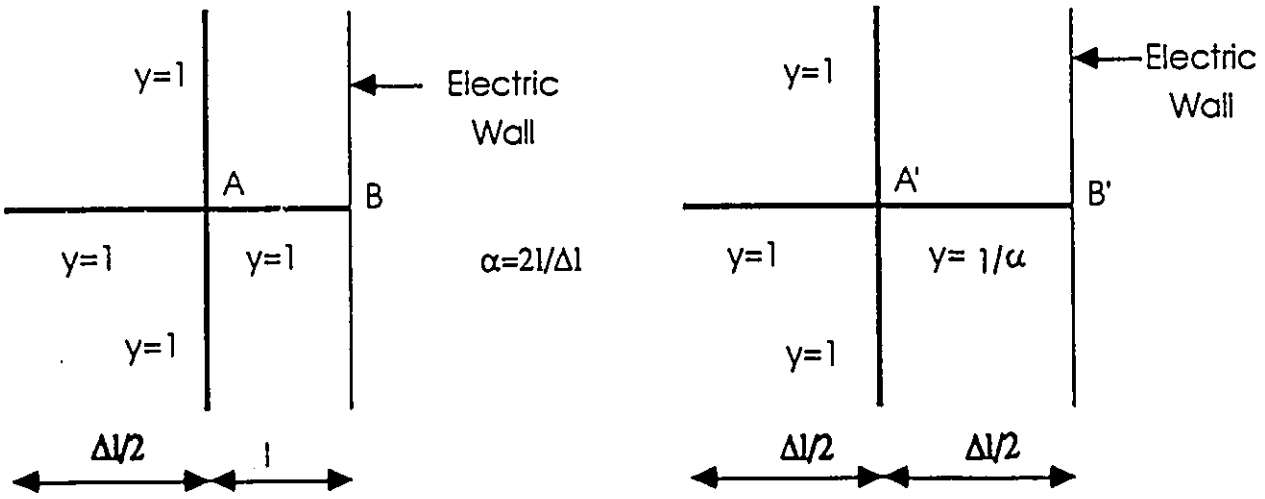

Figure 5.2: Modification of boundary node for arbitrary position of boundary

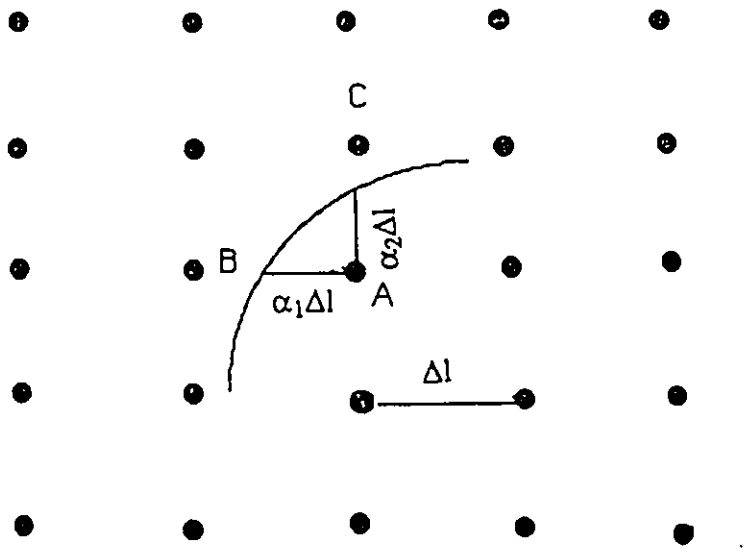

Figure 5.3: Modification of curved boundary

know the length of each branch connecting to the boundary. For example, at node A the characteristic admittance on line AB and AC are $y_1 = 1/\alpha_1$ and $y_2 = 1/\alpha_2$, respectively. Node A in Fig.5.3 is modeled by applying these values into equation (5.6).

Using this technique, results have been obtained for empty symmetrical circular waveguide[42]. Table 1 shows the results for the normalized cutoff wavelength λ_c/a for various modes, both for the piecewise straight-line boundary as discussed in the previous section and arbitrary position of boundary. The radius of this circular waveguide $a = 5\ \Delta\ l$, 500 iterations steps were used. It can be found that the numerical results obtained by latter technique are closer to theoretical results than those obtained by piecewise straight-line boundary modelling. Note that, in both cases, the worst errors for a given family of modes occur for large a/λ. This is due to the dispersion of the TLM network as discussed in chapter 3. This error increases with frequency, the upper bound on the error being as high as 8% for the E_{22} circular mode.

5.4 Reflection Coefficient Modification Technique

In the above section, we discussed the concept of arbitrary wall positioning. A boundary branch which has a length different from $\Delta l/2$ is simply replaced by an equivalent branch of length $\Delta l/2$ having an identical input admittance. This ensures synchronism but requires a different characteristic admittance for the boundary branch and hence, a modification of the impulse scattering matrix of the boundary nodes.

When this technique is applied in 3D symmetrical condensed node, the characteristics impedance of the six arms are not equal, As a result, the symmetry is lost and the number of elements in the scattering matrix is substantially increased. For example, there are sixteen unknowns to be determined, as compared to four unknowns in the case of symmetrical condensed node when the length of only one arm is different from the others. If there are more than two arms connected to the boundary and having different length with any other, the numbers of matrix elements will be at least tripled. These unknown elements are obtained by solving a set of non-linear, second order equations, which is not always well-behaved. Therefore, the technique used in 2D-TLM brings some difficulties when applied to the 3D-condensed node TLM.

Hoefer[42] proposed a technique to account for arbitrary positions of boundaries in 3D-TLM symmetrical condensed node. While the scattering matrix of the boundary nodes remain unchanged, the boundary at the arbitrary distance is represented by an equivalent inductance or capacitance. The differential equation governing the relation between voltage and current in this element is solved in a discretized manner to be compatible with the TLM algorithm. This results in a recursive reflection algorithm at the boundary.

Fig.5.4 shows the representations of short or open circuits by their corresponding induc-

tance or capacitance which are found from their transformed impedance at $\Delta l/2$. Assume that we wish to position a reflecting boundary(electrical or magnetic wall) at a distance $\Delta l/2+l$ from node as shown in Fig.5.4, where l is an arbitrary shift in the boundary position beyond the standard distance $\Delta l/2$ for which the node scattering matrix has been defined. The normalization input impedance at A for a short-circuited transmission-line(electric wall)and its equivalent inductance is:

$$Z_i = jtan\beta l = \frac{j\omega L}{Z_0} \tag{5.7}$$

For open-circuited transmission-line (magnetic wall),one has:

$$Z_i = \frac{1}{jtan\beta l} = \frac{1}{j\omega C Z_0} \tag{5.8}$$

Since the mesh size is small enough to comply with the dispersion requirement ($\Delta l/\lambda < 0.05$), i.e., $\beta l << 1$, the inductance and capacitance can be considered independent of frequency ($tan\beta l \simeq \beta l$), yielding:

$$L = \frac{Z_0 l}{c} \tag{5.9}$$

$$C = \frac{l}{cZ_0} \tag{5.10}$$

where c is the propagation velocity in the TLM mesh.

It is now possible to write the differential equation governing voltage and current of the input of the reactive stubs in terms of the incident and reflected impulses, and to replace this differential equation by a difference equation. Consider a simple inductance circuit as shown in Fig.5.5. The differential equation that governs the circuit is:

$$V = L\frac{di}{dt} \tag{5.11}$$

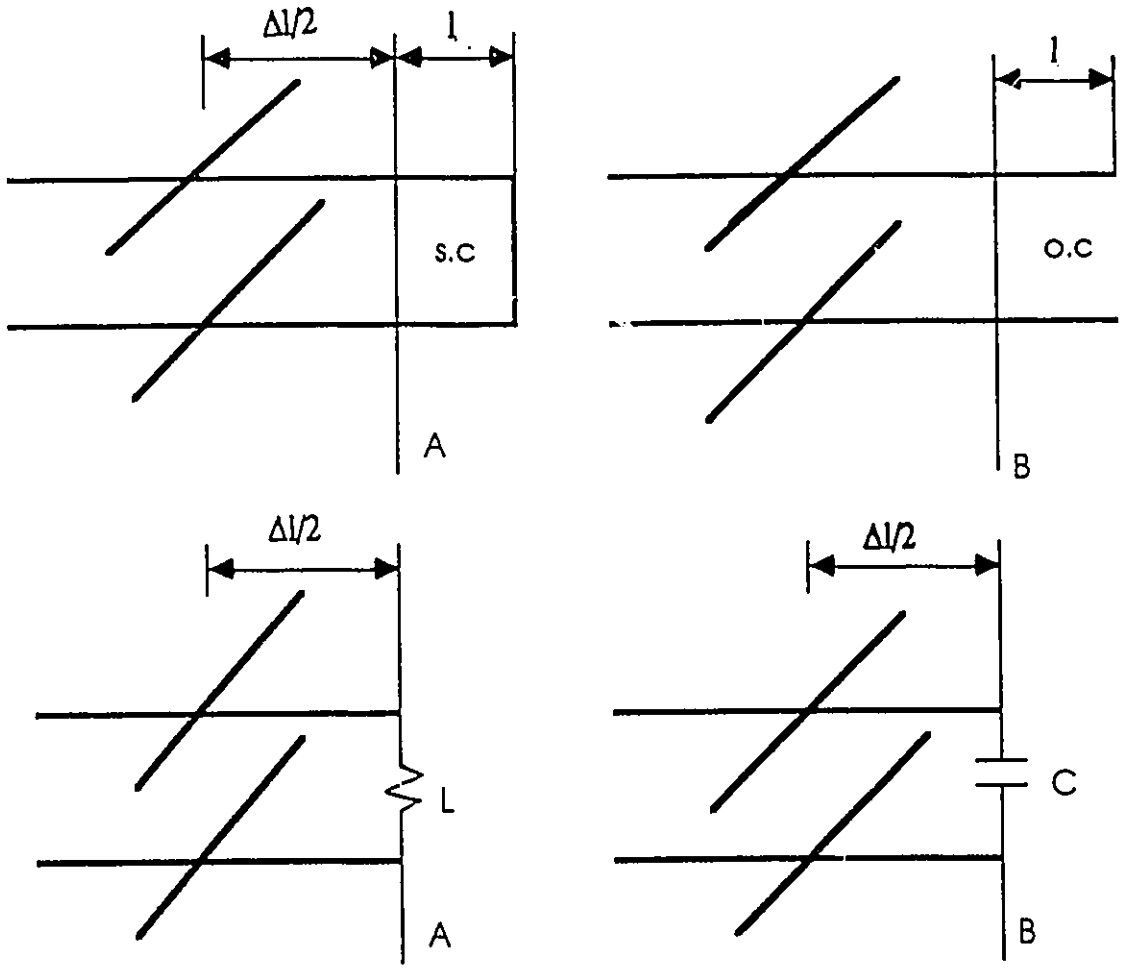

Figure 5.4: Extension of short and open-circuit boundaries in a TLM mesh, and their representation by equivalent reactances

The corresponding difference equation is:

$$ {}_kV^r + {}_k V^i = L\frac{{}^k i - {}_{k-1} i}{\Delta t} = L\frac{\frac{{}_kV^r - {}_kV^i}{Z_0} - \frac{{}_{k-1}V^r - {}_{k-1}V^i}{Z_0}}{\Delta t} \tag{5.12} $$

After manipulating equation (5.12), one has:

$$ {}_kV^i = (\frac{\kappa - 1}{\kappa + 1}){}_kV^r + \frac{\kappa}{\kappa + 1}(-{}_{k-1}V^r + {}_{k-1} V^i) \tag{5.13} $$

where subscripts k and $k-1$ represent the voltage values at time $k \Delta t$ and $(k-1) \Delta t$, respectively. κ is equal to $2l/\Delta l$ in the 3D-TLM case, and $\sqrt{2}l/\Delta l$ in the 2D TLM case.

Similar analysis is applied to the equivalent capacitance circuit and the following equation is obtained:

$$ {}_kV^i = (\frac{1 - \kappa}{1 + \kappa}){}_kV^r + \frac{\kappa}{1 + \kappa}({}_{k-1}V^r + {}_{k-1} V^i) \tag{5.14} $$

Equation (5.13) and (5.14) can be written in the following recursive formula:

$$ {}_kV^i = (\frac{\kappa - 1}{\kappa + 1}){}_kV^r + \frac{\kappa}{\kappa + 1}(\rho_{k-1}V^r + {}_{k-1} V^i) \tag{5.15} $$

where $\rho = 1$ for a magnetic wall and $\rho = -1$ for an electric wall. The reflection coefficient between the dielectric interface is $\rho = (\eta_2 - \eta_1)/(\eta_2 + \eta_1)$, where η_1, η_2 are the wave impedances of the corresponding dielectric material. Equation(5.15) indicates that the present impulse reflected from the boundary in the reference plane at $\Delta l/2$ depends on the present incident impulse as well as on the previous incident and reflected ones.

The recursive formula (5.15) is based on the assumption that the distance between the node and the boundary is beyond $\Delta l/2$. If this distance is less than $\Delta l/2$, the same procedure as before, in which the equivalent lumped elements are positioned $\Delta l/2$ from the node, is applied and lead to the following recursive formula:

$$ {}_kV^i = (\frac{\kappa + 1}{\kappa - 1}){}_kV^r + \frac{\kappa}{\kappa - 1}(\rho_{k-1}V^r + {}_{k-1} V^i) \tag{5.16} $$

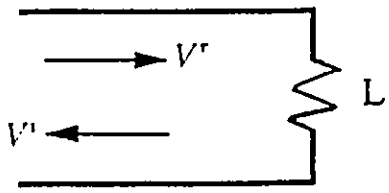

Figure 5.5: **Equivalent inductance circuit**

It has to be noted, in the case described in Fig.5.4 the boundaries can not be exactly at the nodes($l \neq \Delta l/2$). In this case κ=1 and (5.16) become indeterminate.

To alleviate this problem, one can choose a value of l which is very close to $\Delta l/2$, for example, $l = 0.8 \Delta\, l/2$, to approximate the real distance $l = \Delta l/2$. It has been demonstrated that l can be chosen up to $l = 0.92 \Delta\, l/2$ without causing numerical instabilities. This substitution represents a very accurate treatment of the above mentioned boundaries. Furthermore, very few boundaries are located exactly on the half-way between nodes and can be eliminated by rediscretizing the space if necessary.

Although these recursive formulae are derived for straight-line boundary, they can be also applied to curved boundaries that run at arbitrary positions between nodes. This technique is found to be more accurate than staircase approximation when applied to 3D symmetrical condensed node as shown in chapter 7.

Chapter 6

Results and Discussions

6.1 Introduction

In previous chapters, the new TLM algorithm was introduced and its applications to inhomogeneous, lossy and anisotropic materials were described. Special treatment of the curved boundary was discussed. In this chapter, the dispersion characteristics of some rectangular and circular guided wave structures are analysed by using this new TLM algorithm. Since all the structures investigated are assumed to be symmetrical, only half of the structure is needed to obtain the total dispersion characteristics. The simulation results are compared with theoretical solutions or those provided by other proved numerical methods in open literatures.

6.2 System Size and CPU Time

The new TLM algorithm requires that $12 \times N_1 \times N_2$ mesh for modelling of homogeneous material and $18 \times N_1 \times N_2$ mesh for inhomogeneous material be discretized, N_1, N_2 being the number of mesh along horizontal and vertical directions. Theoretically, N_1, N_2 are chosen as large as possible to obtain the small value of $\Delta l/\lambda$ so that mesh dispersion can be neglected. But the maximum mesh size can not exceed a certain value without

exhausting computer resources. For example, the maximum value for the products of N_1 and N_2 can be 5284 (72×72) for modelling of homogeneous material and 3456 (58×58) for inhomogeneous materials on TRIX 3100 workstation. Any products of N_1 and N_2 greater than these values will exhaust computer resourses. It is found that more than five minute CPU-time is taken for 500 iterations and around twenty minutes for 3000 iterations under maximum mesh size. The same program was run on the University of Toronto Cray XM-P supercomputer for the 72×72 mesh size and 3000 iterations. It was observed that the total computation time was only 3 seconds. This represents a drastic improvement of nearly 3 orders of magnitude. This demonstrates the suitability of the TLM algorithm for such machine.

6.3 Rectangular Guided Wave Structures

6.3.1 Empty Waveguide

A simple empty waveguide is used to validate the new TLM algorithm since an analytical solutions is available. The dispersion characteristic can be found in any textbook and is given by:

$$\beta = \sqrt{4\pi^2 f^2 \mu\epsilon - [(\frac{m\pi}{a})^2 + (\frac{n\pi}{b})^2]^2} \tag{6.1}$$

where β is the phase constant. a and b are waveguide dimensions along horizontal and vertical directions, respectively, and m, n are mode indexes. Fig.6.4 and Fig.6.5 show the numerical results for the first mode(TE_{10}) and the second mode (TE_{20}) (3000 iteration times, 72×72 mesh), respectively. By comparing with corresponding theoretical results, it is obvious that the numerical soiutions for the first mode agree with the theoretical solutions better than the second mode. This illustrates that the effect due to dispersion of the TLM mesh increases with frequency.

Fig.6.6 shows the results for different number of iterations. It is found that the dispersive curves (TE_{10} mode) differ from the theoretical results as the number of iterations goes down. It was observed that 3000 iterations are large enough under maximum mesh discretization to maintain a very small truncation error while keeping a reasonable computation time. Therefore, all the simulations followed are based on 3000 iterations and maximum mesh discretization unless otherwise specified.

6.3.2 Lossy Material-filled Waveguide

It has been demonstrated that the new TLM algorithm works well for empty waveguide in above section. Its theoretical application to lossy material are the same as empty waveguide except that some modifications of scattering matrix have to be done. Here, we consider dielectric losses although a similar procedure can be followed for magnetic losses. The dielectric constant is assumed to be independent of frequency. Fig.6.7 shows the dispersion curve for the first mode(TE_{10}) of a lossy waveguide, where ($a = 6cm, b = 4cm$). 89 × 59 meshes is employed to account for the dimension difference along horizontal and vertical directions. It is found that the differences between this method and FEM[35] are within 3% provided that $\Delta l/\lambda$ is less than 0.05.

6.3.3 Inhomogeneous Waveguide

The effects of inhomogeneous material can be modelled by introducing the extra stubs. The structure investigated is a dielectric-slab-loaded waveguide as shown in Fig.6.8. As the wave propagates in the waveguide, most of the energy flow occurs in the dielectric-slabs. Therefore, the output point should be located in the dielectric material in order that field level is high enough to be extracted and then the peak values in Fourier domain response can be clearly observed. In this case, the excitation point is chosen to be close to the right

hand of the left dielectric slab while the output point is very close to the left hand of the right dielectric slab for the first mode. Those for the second mode have to be away from the center of the waveguide and within the dielectric, since the peak value of the field is around there.

Since extra stubs are involved in the modelling of dielectric material, the number of meshes has to be reduced to 58 × 58 in order to remain within the computing capabilities. Although the mesh dispersion for the modelling of inhomogeneous material is slightly larger than homogeneous material[8], good results are still achieved by comparing with those obtained by Mode Matching Method[36] as shown in Fig.6.8, where the phase constant is normalized to that of free space. The results by both methods start to differ from each other when frequency goes over $25GHz$ due again to the dispersion produced by the TLM mesh.

6.3.4 Shielded Dielectric Wave Guided

Since there are surface wave outside the dielectric waveguide, this constitutes an open problems. For simplicity, all the dielectric waveguides investigated are shielded to limit the computation domain as done in open literatures[34][35].

First, we consider a dielectric waveguide as shown in Fig.6.9, where n_1 and n_2 are the refractive indices of the core and cladding regions, respectively. Assuming the shielded boundary is five times larger than dielectric waveguide($X = 10l, Y = 5l$) and considered to be large enough without significantly disturbing the fields since they decay very rapidly outside the dielectric. The original open problem is now substituted by a bounded one without complicated absorbing boundary algorithm.

Fig.6.9 shows the propagation characteristics of the first mode(E_{11}) of this dielectric

waveguide. The normalized frequency v and the normalized propagation constant b are defined as:

$$v = \beta_0 t\sqrt{n_1^2 - n_2^2}/\pi \tag{6.2}$$

$$b = \frac{(\beta/\beta_0)^2 - n_2^2}{n_1^2 - n_2^2} \tag{6.3}$$

Good agreement is observed with FEM[34] and empirical solutions[37].

Next example is an equilateral triangular core waveguide as shown in Fig.6.10. The refractive indices of core and cladding regions are n_1=1.5 and n_2=1.0, respectively. The shield dimension is chosen to be $X = 6t$ and $Y = 5t$. Our results agree well with those presented by Koshiba[34] and Yeh[38] at low frequencies. The curve produced by the TLM method differs at high frequencies because of the mesh dispersion, as mentioned before.

Up to now, we have analysed the propagation characteristics of some simple guided wave structures. A more elaborated application of this new TLM algorithm is a lossy image guide shielded with a perfectly conducting box. Fig.6.11 shows the dispersion characteristics for E_{11} mode as well as the geometry of this image waveguide in which the complex relative permittivity is chosen as $\epsilon = 1.5 - j1.5$, $a = 4t, b = 2t$.

Another application involves the modelling of anisotropic material. Consider a dielectric waveguide which is composed of lossy anisotropic material. Fig.6.12 shows the propagation characteristics for shielded image guide composed of lossy anisotropic dielectric. The real part of dielectric is assumed to be anisotropic. Two cases are investigated. First, the real part of the dielectric is ϵ_y'=2 and ϵ_y'=1.5 for the second case. The dielectric constants along x and z are considered equal ($\epsilon_x = \epsilon_z = 1.5 - j1.5$). If the imaginary part of dielectric is anisotropic, the simulation program is not modified because the computer program handles loss tangent as input data.

The above two applications are compared with those produced by FEM[35] as shown in Fig.6.11 and Fig.6.12, respectively. The maximum difference is 4% in both cases.

Finally, we analyse a shielded microstrip transmission as shown in Fig.6.13, where $w = 1.27mm, t = 0, h = 1.27, X = 12.7mm$ and $Y = 12.7mm$. The microstrip is located on a isotropic substrate with ϵ_r=8.875. It has to be noted that the microstrip is made of metal and the field around the microstrip edges is near singular. More meshes have to be placed around the microstrip to account for this quasi-singularity. As a result, 25×25 meshes are devoted to the area around the microstrip among 58×58 meshes. This indicates that more than 40% of the meshes contributes to the areas near edges, while only 20% contributes to the remaining region. This variable mesh discretization is shown in Fig.6.1 together with regular mesh discretization used before. The simulated results by using regular mesh size and variable mesh size, as described before, are shown in Fig.6.13. Solutions produced by the variable mesh size technique are closer to FEM[34], which shows the importance of accounting for conducting edges or corners in structures.

Fig.6.14 shows the propagation characteristics for the first and the second mode by using variable mesh size technique. Results agree well with those obtained by FEM[34]. The difference at high frequencies is due to the mesh dispersion of the TLM method. Again, it should be noted that accuracy can be improved by the use of large computer on which more memory is available.

6.3.5 Open Guided Wave Structures

There are cases in which the microstrip transmission line is not shielded. This becomes an open problem. Absorbing Boundary Conditions(ABC) have to be employed to limit the computation domain. The ABC introduced in section 4.4 of chapter 4 is used to investigate

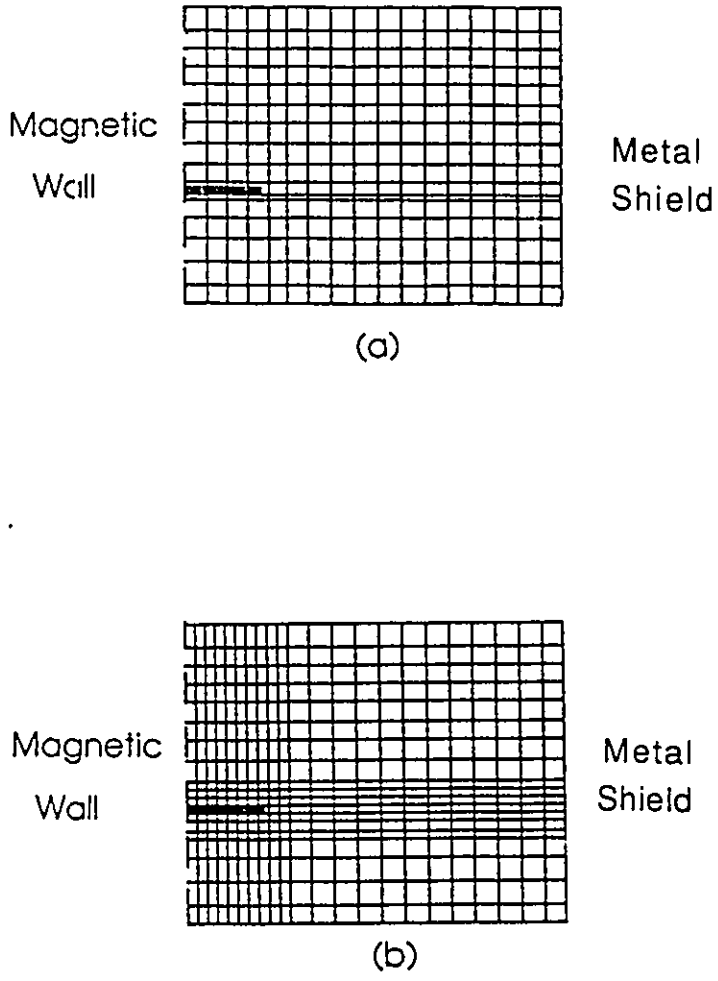

Figure 6.1: Regular mesh discretization(a) Variable mesh discretization(b)

how large the dimension should be to obtain accurate simulation results.

Fig.6.15 shows TLM results for various dimensions of ABCs. The parameters used for the computations are $h = 3.17mm, w/h = 0.96, \epsilon_r = 11.7, \mu_r = 1.0$. The effective dielectric constant $\epsilon_{reff}(\omega)$ are used to account for the dispersive characteristics of the microstrip. It is defined as[39]:

$$\epsilon_{reff}(\omega) = \frac{\beta^2(\omega)}{\omega^2 \epsilon_0 \mu_0} \tag{6.4}$$

The dimension of ABC-1 is chosen to be $X = Y = 10mm$ as shown in Fig.6.15. The variable mesh size technique is used for its better accuracy, as shown before. More than 60% of the total available meshes(36×36) contributes to areas around edges of the microstrip. As shown in Fig 6.2, there is quite a large difference between dispersive curves produced by FDTD[39], empirical formula by Pramanick[40] and the proposed TLM method. This is because the absorbing wall is placed too close to the microstrip and the field does not impinge normally on the absorbing wall and, hence, does not give the zero reflection coefficient at the absorbing boundary as explained in[27]. At high frequencies, this difference is even much higher because of the large mesh dispersion of TLM method. As the absorbing boundary is moved further(for ABC-2, $X = Y = 18mm$, for ABC-3, $X = Y = 30mm$), the level of the field that does not perpendicularly impinge on the boundaries is very small and, therefore, has very little effect on the modelling results.

6.4 Circular Guided Wave Structures

6.4.1 Empty Waveguide

The empty circular waveguide is used to validate the two boundary treatment techniques as introduced in chapter 5 since its exact solutions can be obtained. Theoretical dispersion

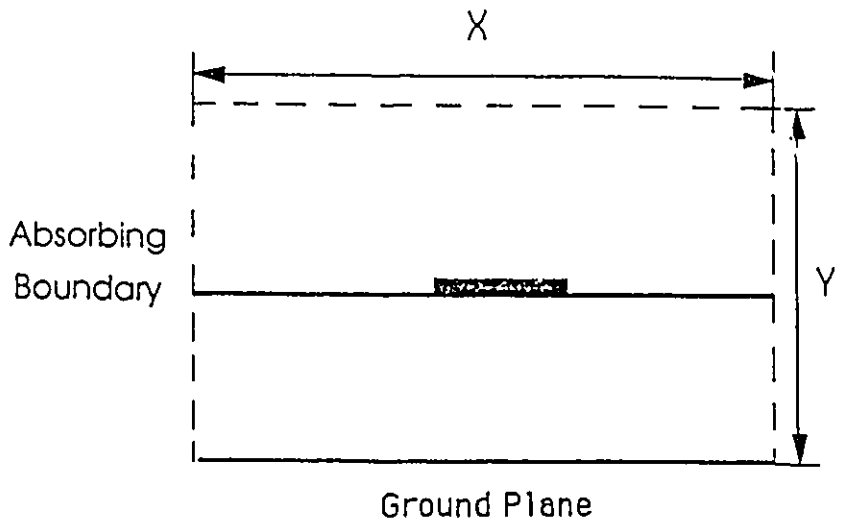

Figure 6.2: The layout of absorbing boundary for open microstrip transmission line

equation of empty circular waveguide for TM modes is:

$$\beta = \sqrt{\omega^2 \mu_0 c_0 - (\frac{p_{mn}}{a})^2} \tag{6.5}$$

and for TE modes:

$$\beta = \sqrt{\omega^2 \mu_0 c_0 - (\frac{p'_{mn}}{a})^2} \tag{6.6}$$

Where a is the radius of circular waveguide p_{mn} and p'_{mn} are the roots of the first kind Bessel function and the derivative of the first kind Bessel function, respectively. m,n are mode numbers.

Two cases are considered to demonstrate that reflection coefficient modification technique is more accurate than the staircase approximation. First, a empty circular waveguide with radius $a = 5mm$ is discretized by 8×8 meshes($\Delta l/\lambda$=0.2). This represents a very coarse meshes. Fig.6.16 shows the comparison between dispersion curve of TE_{10} obtained by the use of piecewise boundary approximation, reflection coefficient modification technique and exact solution. It is found that the solution produced by using reflection

coefficient modification technique is closer to the exact one than piecewise boundary approximation.

For the second case, the circular waveguide is discretized under maximum mesh size as stated before. More accurate results produced by reflection coefficient modification technique are still observed as shown in Fig.6.17 and 6.18. The difference between the two techniques and theoretical solution is smaller than previous one since more meshes is employed and, hence, mesh dispersion is smaller.

It is also observed from Fig.6.17 and 6.18 that the first mode(TE_{11}) agrees better than the second mode(TM_{01}) because the mesh dispersion is smaller at lower frequencies. It has to be noted that the excitation and output points for TM_{01} mode should be placed around the center since the electric field exhibits the maximum value in the center of the waveguide.

6.4.2 Circular Dielectric Guided Wave Structures

In above sections, the shielded boundary around the rectangular guided wave structures is used to limit the computation domain simply because we want to compare the simulation results with those by other method in open literatures. Here, we applied absorbing boundaries around the circular dielectric guided wave structure to limit the computation domain. Fig.6.19 illustrates the dispersion characteristics of dielectric circular waveguide for the first mode(HE_{11}) by using different boundary treatment techniques. The dimension of the absorbing boundary is $X = 4a, Y = 4b$ as shown in Fig.6.3. It is demonstrated that better results are achieved under this absorbing boundary dimension $X = 4a, Y = 4b$. The location of excitation and output points are similar to those of TE_{11} mode since it is a major component of the hybrid of HE_{11} mode. Once again, it is proved that the

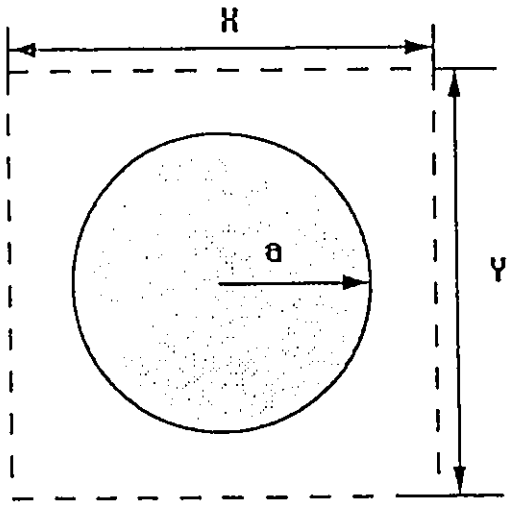

Figure 6.3: The layout of absorbing boundary around the dielectric circular waveguide

reflection coefficient modification technique is more accurate than piecewise boundary approximation. Errors come from the imperfect absorbing boundary and mesh dispersion, particularly at high frequencies.

As a more complex application, consider an inhomogeneous circular dielectric waveguide as shown in Fig.20. The dielectric constant of the upper half and lower half is $\epsilon_1 = 2.56$ and $\epsilon_2 = 2.03$, respectively. Fig.6.20 shows dispersion characteristics of the second mode($_0TE_{01}$), where $_0TE_{01}$ is not complete transverse mode but hybrid mode in which TE_{01} mode is a dominant component. The same dimension of absorbing boundary, as used in the case of homogeneous dielectric waveguide, is applied. Simulation results are compared with those obtained by point-matching method[10] and agree well especially at low frequency.

The last application of the proposed TLM algorithm is the case of bilateral finlines in circular waveguide housing as shown in Fig.6.21. For reasons stated before, a denser mesh is used in areas around the conduction strip edges Reflection coefficient modification technique is used to deal with the circular boundary. Fig.6.21 shows the dispersion curves

for HE_{11} mode, where $a = 4.615mm, h = 0.254mm, w = 0.3mm, \epsilon_r = 2.2$. Good agreement is observed with FEM[41].

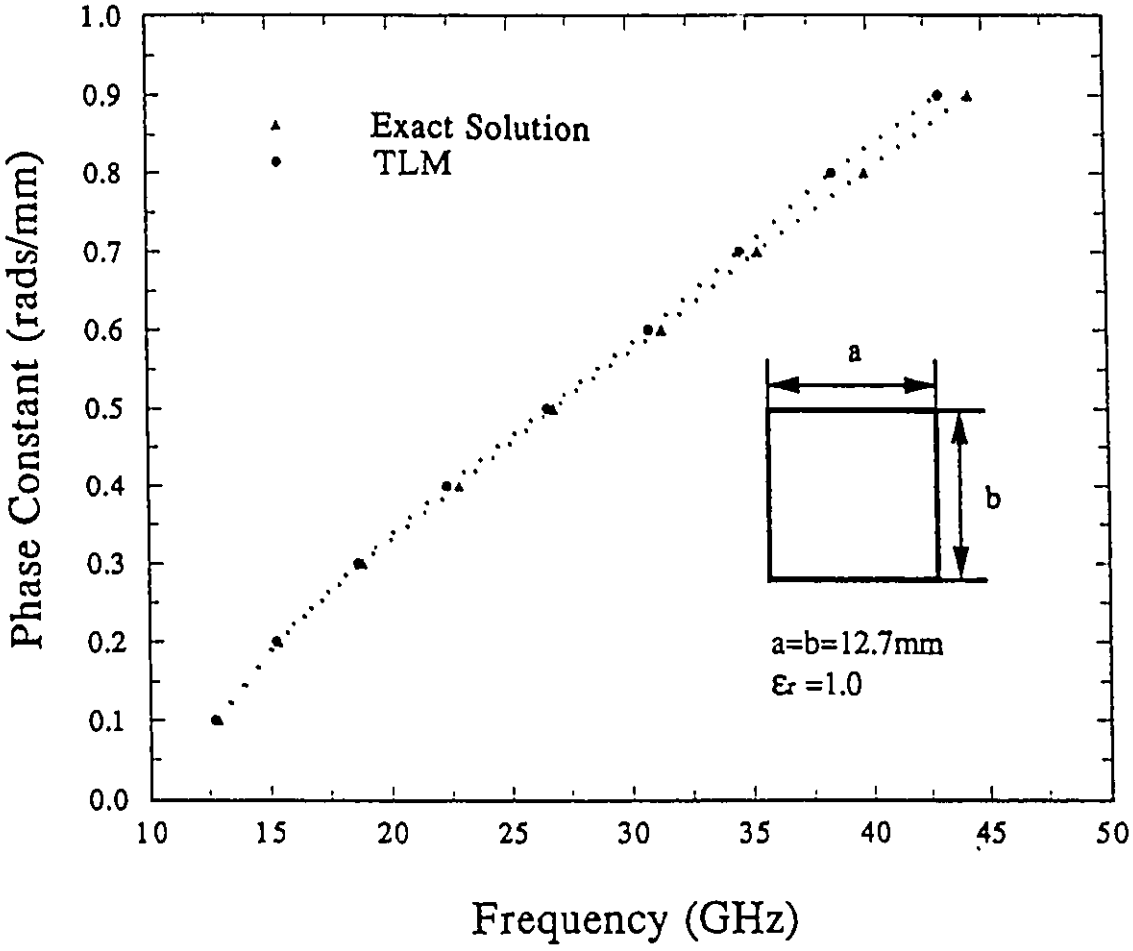

Figure 6.4: Dispersion characteristics of TE_{10} mode for empty waveguide (a =b = 12.7mm, $\epsilon_r = 1.0$)

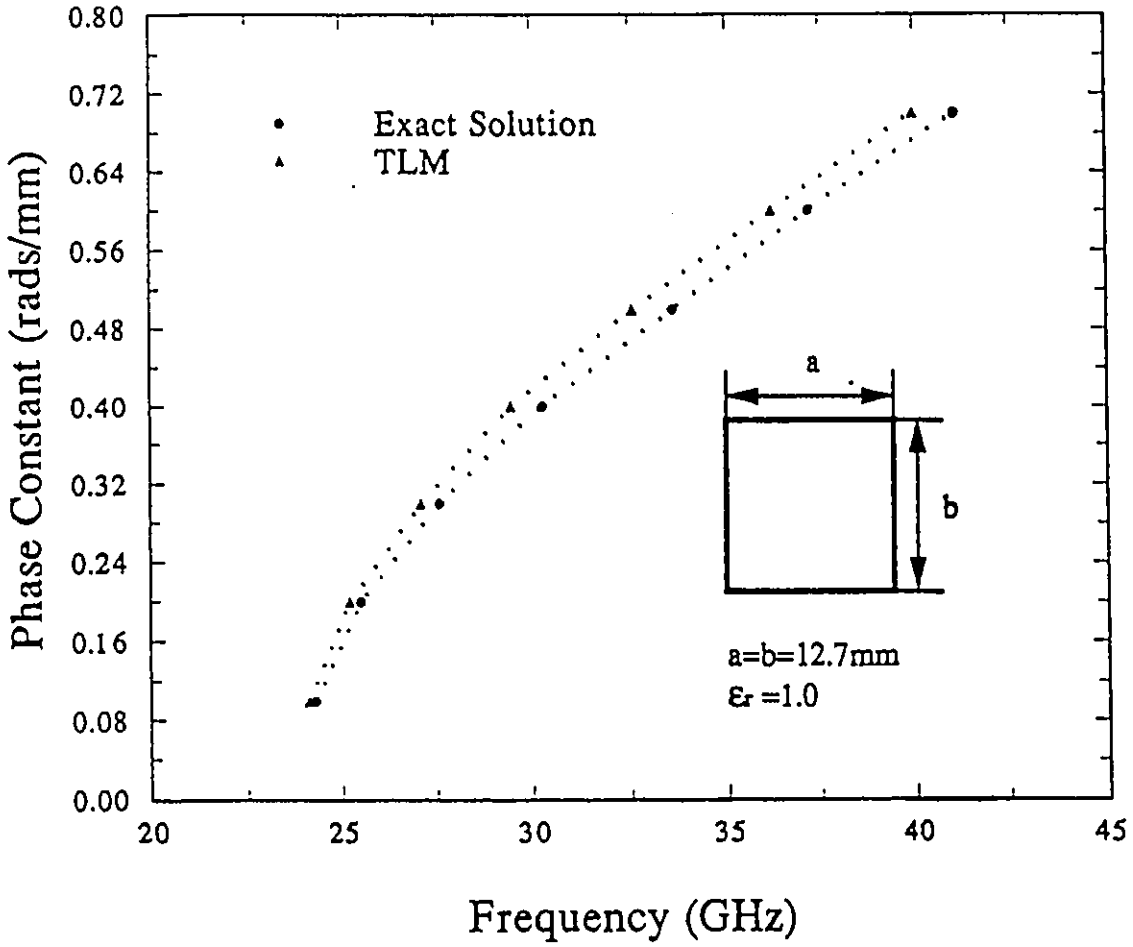

Figure 6.5: Dispersion characteristics of TE_{20} mode for empty waveguide (a =b = 12.7mm, $\epsilon_r = 1.0$)

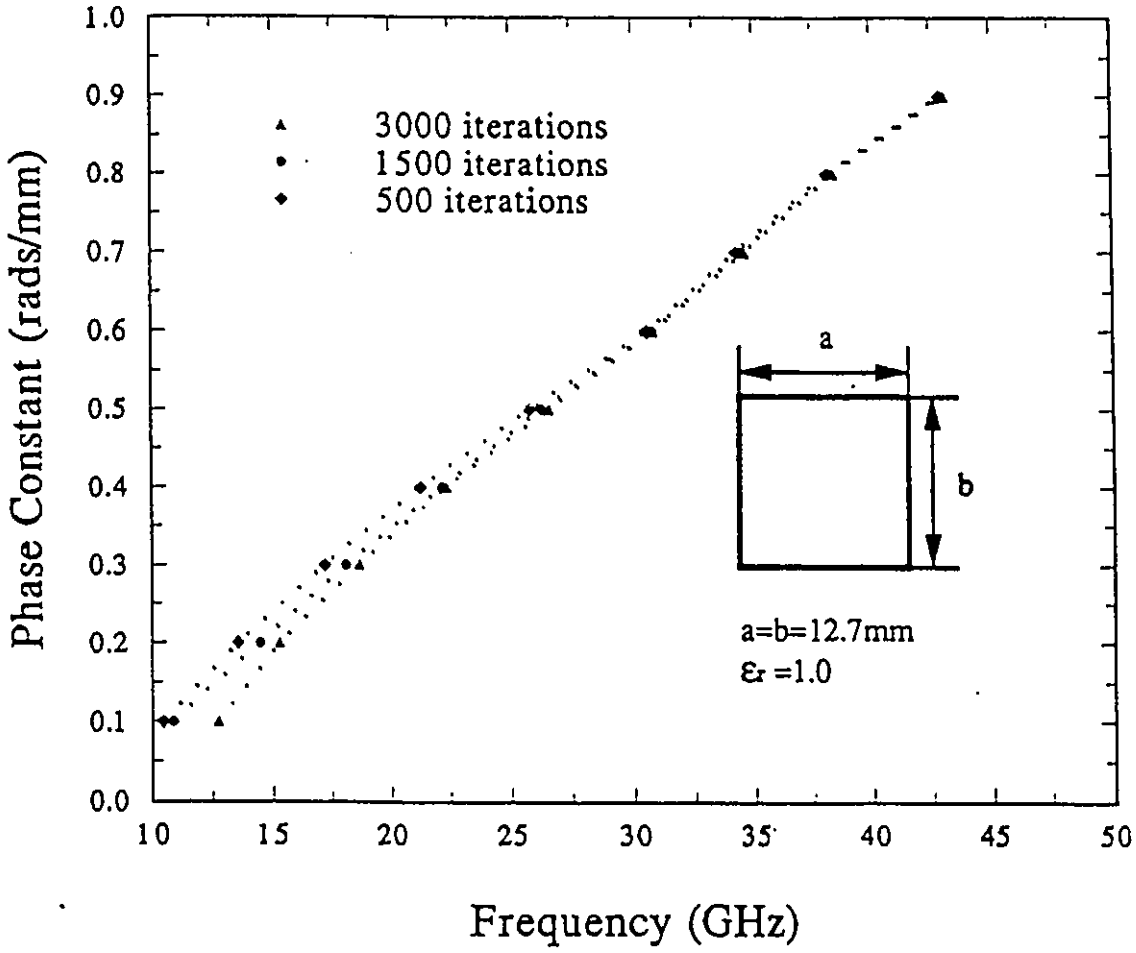

Figure 6.6: Comparison of dispersion characteristics of TE_{10} mode of different iteration times (a=b=12.7mm,ϵ_r = 1.0)

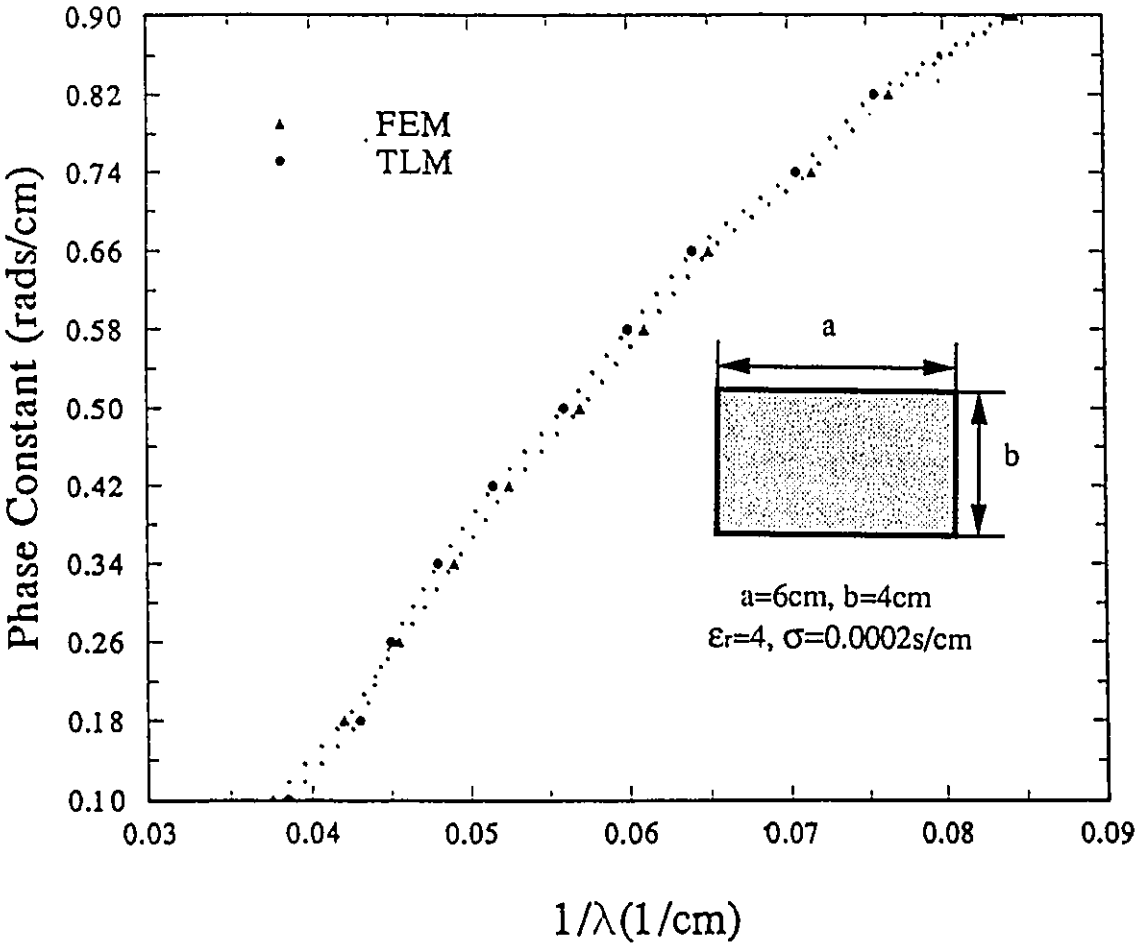

Figure 6.7: Dispersion characteristics of TE_{10} mode for lossy waveguide (a=6cm, b=4cm, $\epsilon_r = 4.0, \sigma = 0.0002 s/cm$)

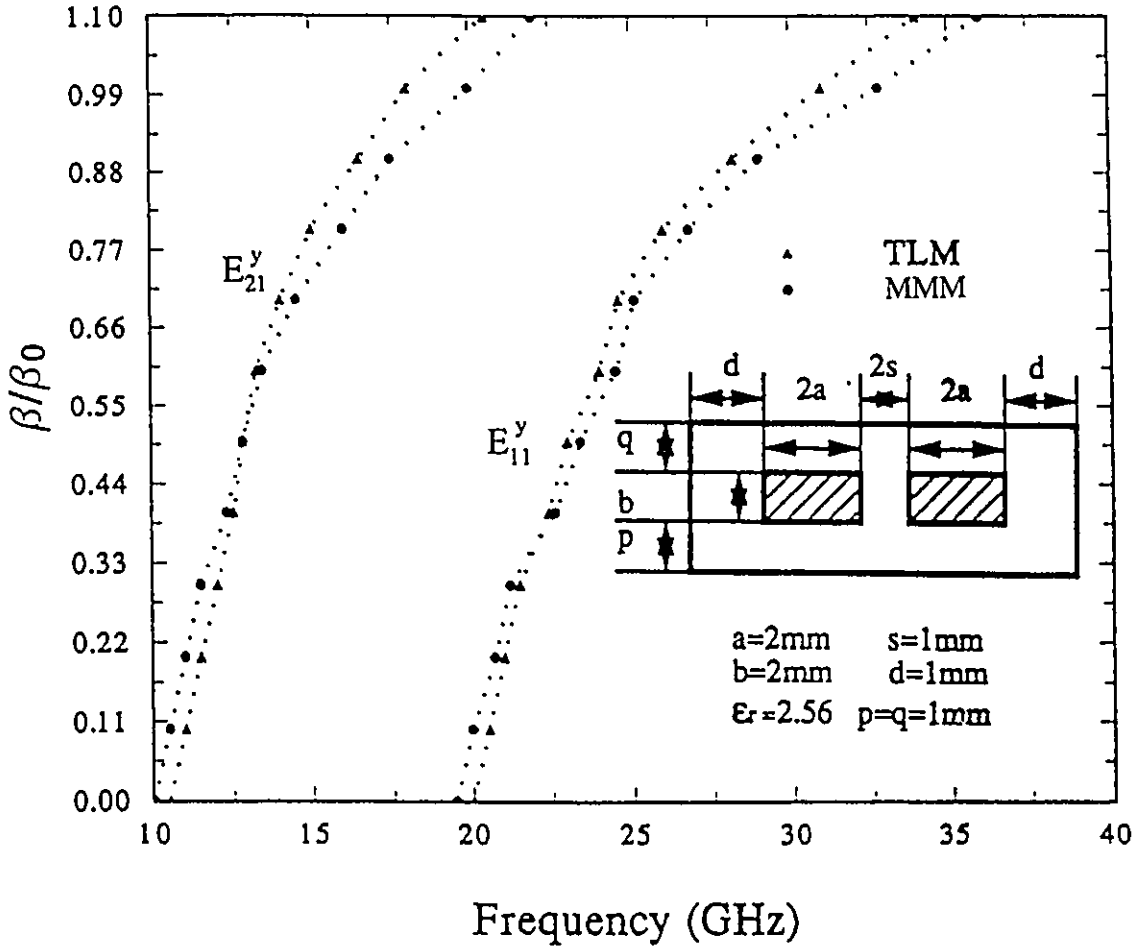

Figure 6.8: Dispersion characteristics of the first and the second modes for dielectric-slab-loaded waveguide (a=2mm, s=1mm, b=2mm, d=1mm, p=q=1mm, $\epsilon_r = 2.56$)

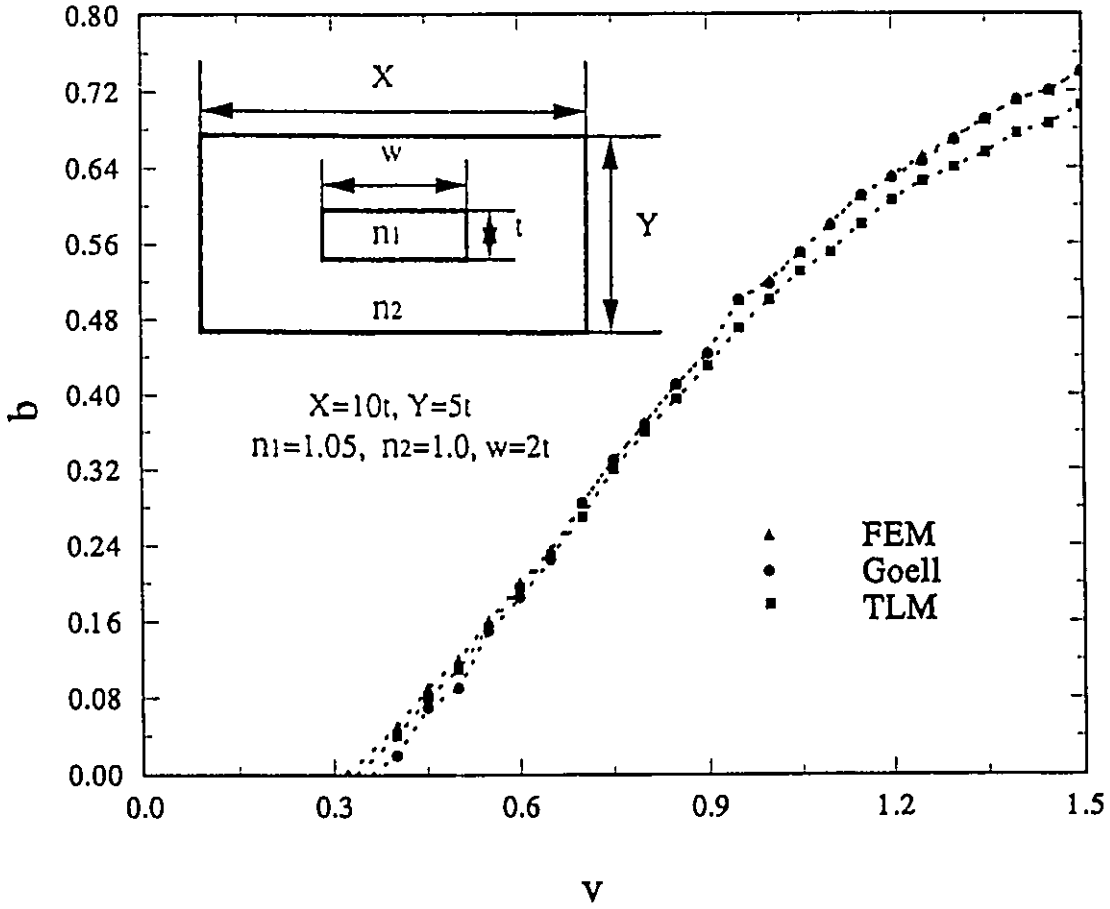

Figure 6.9: Dispersion characteristics of the first mode(E_{11}^x) of dielectric rectangular waveguide ($n_1 = 1.05, n_2 = 1.0, X = 10t, Y = 5t, w = 2t$)

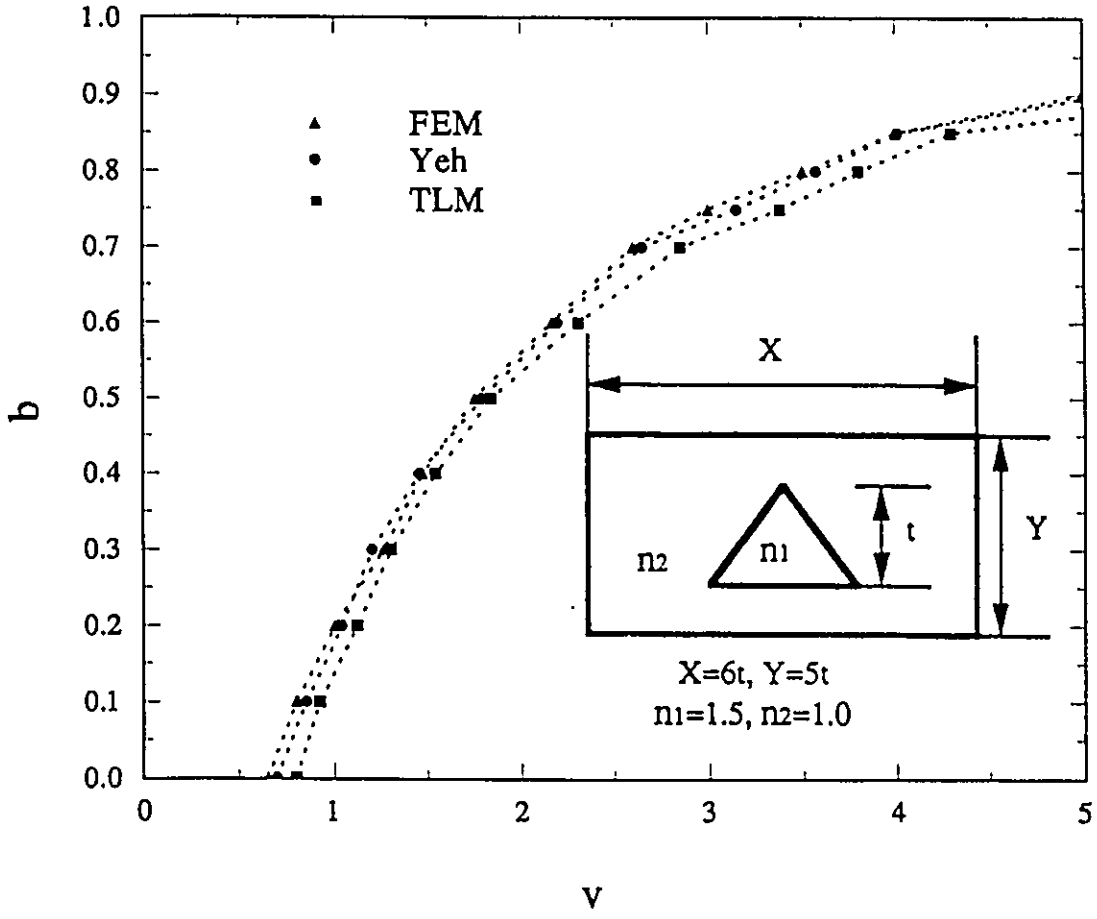

Figure 6.10: **Dispersion characteristics of the first mode(E_{11}^{y}) of an equilaternal triangular core waveguide ($n_1 = 1.5, n_2 = 1.0, X = 6t, Y = 5t$)**

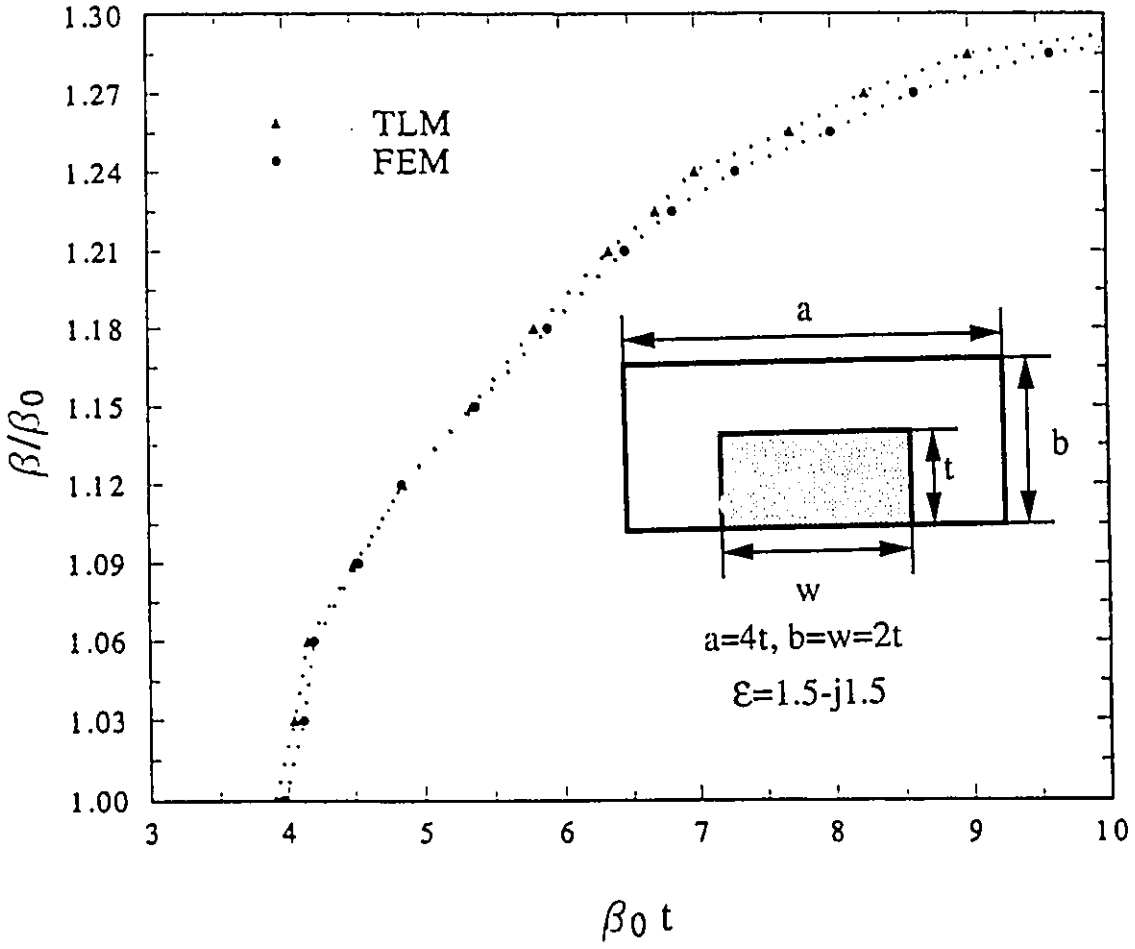

Figure 6.11: Dispersion characteristics of the shielded image guide composed of lossy isotropic dielectric (E_{11}^{y})(a=4t,b=w=2t,ϵ=1.5-j1.5)

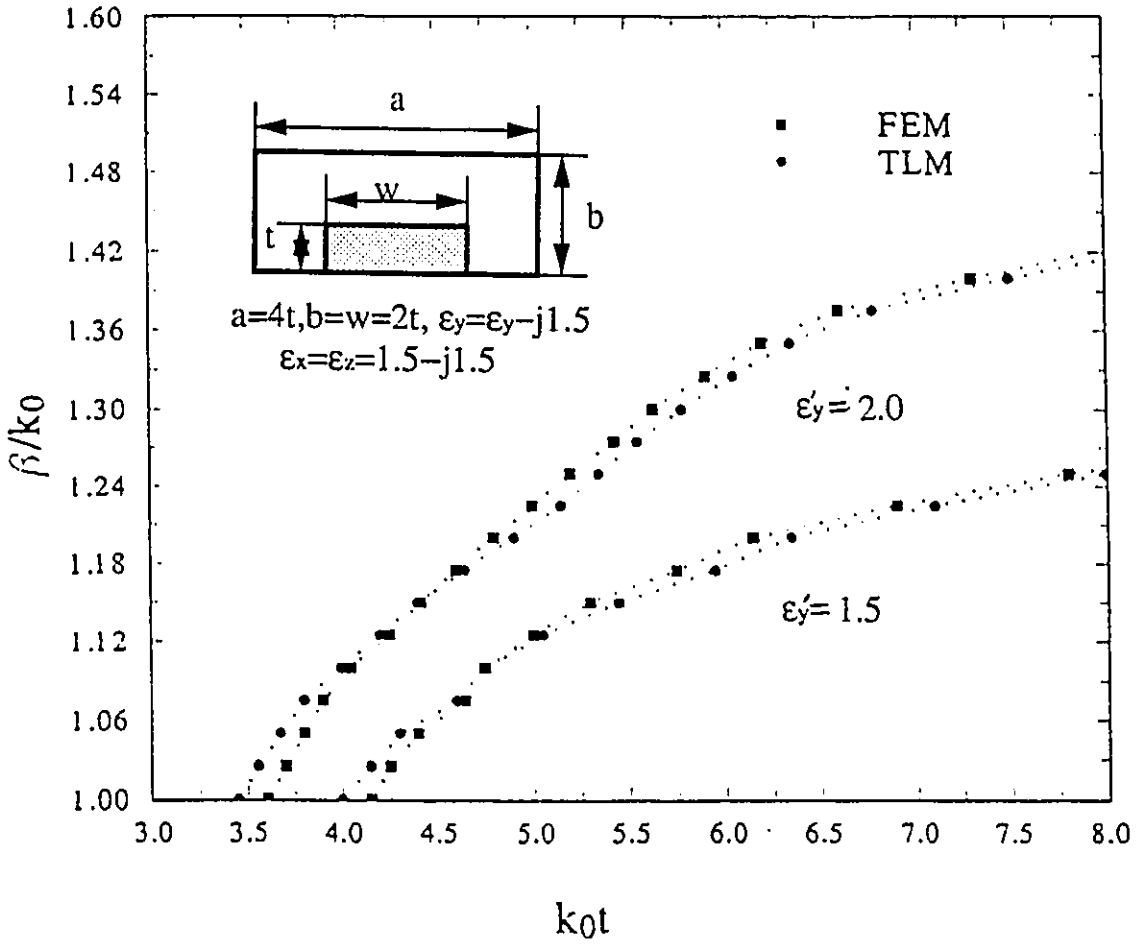

Figure 6.12: Dispersion characteristics of the shielded image guide composed of lossy anisotropic dielectric(E_{11}^{y}),the real part of the dielectric is assumed to be anisotropic(a=4t,b=w=2t,$\epsilon_x = \epsilon_z = 1.5 - j1.5, \epsilon_y = \epsilon_y' - j1.5$)

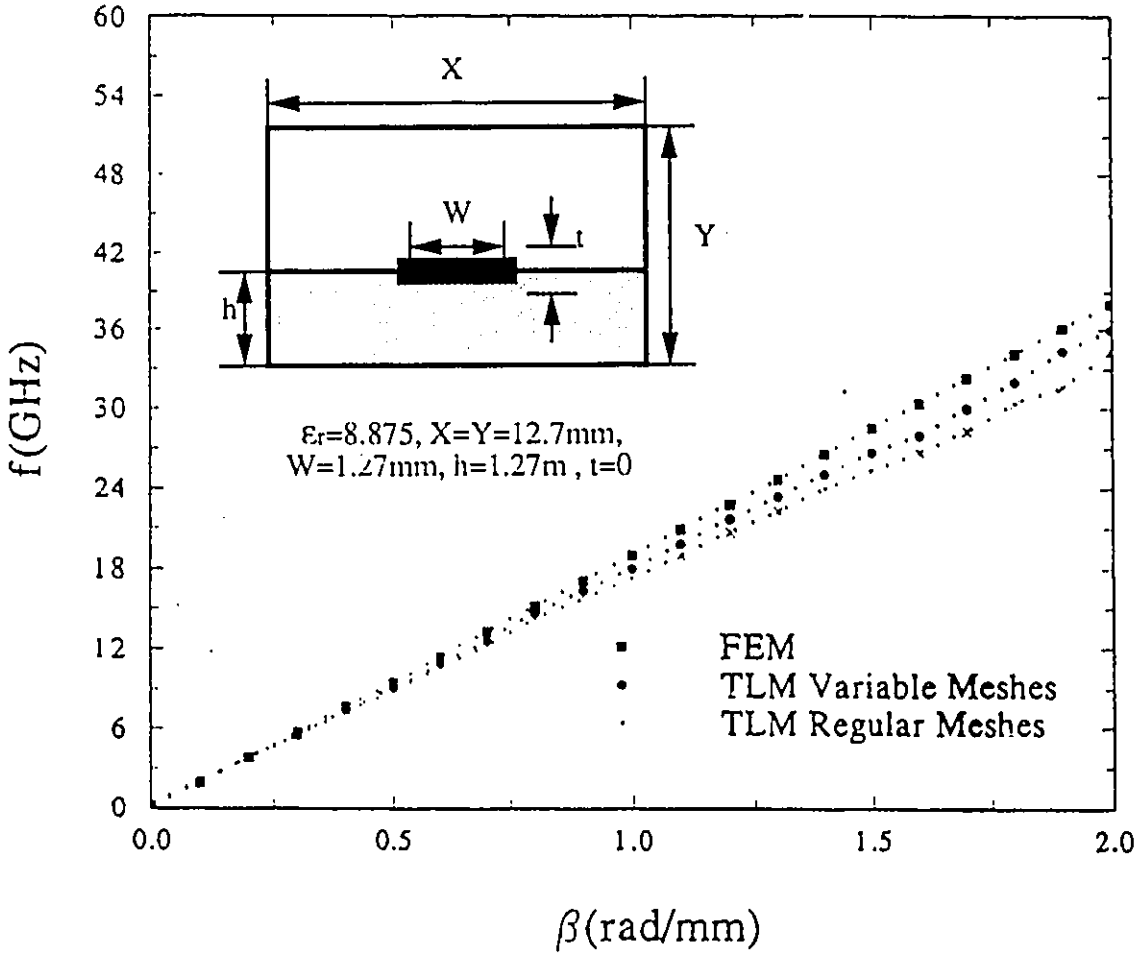

Figure 6.13: Comparison of variable and regular mesh discretization for the first mode (ϵ_r=8.875,X=Y=12.7mm,w=h=1.27mm,t=0mm)

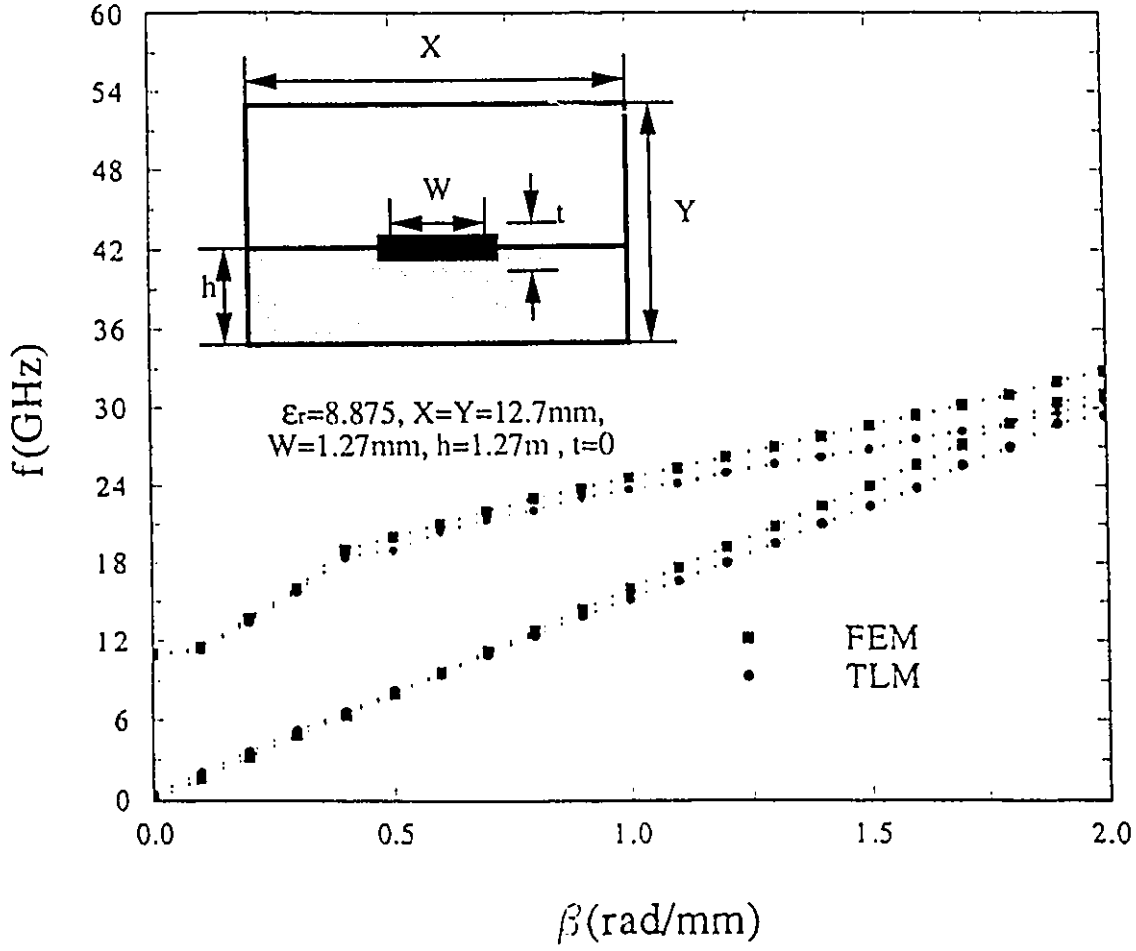

Figure 6.14: Dispersion characteristics of a shielded microstrip transmission line for the first and second modes (ϵ_r=8.875, X=Y=12.7mm, w=h=1.27mm, t=0mm)

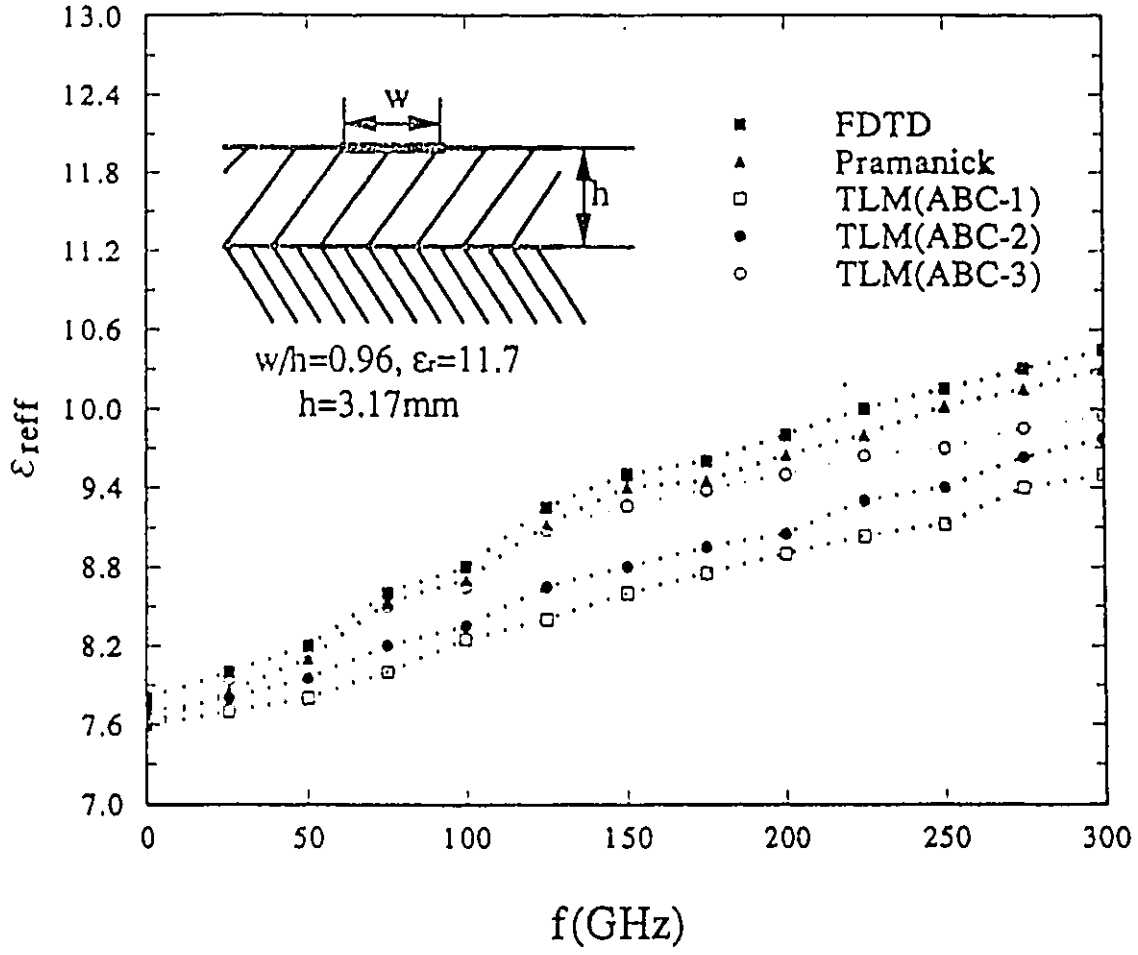

Figure 6.15: Comparison of effective constant ϵ_{reff} as computed by FDTD, empirical formula and different dimensions of the third order absorbing boundary (ϵ_r=11.7,w/h=0.96, h=3.17mm,t=0,μ_r = 1.0),

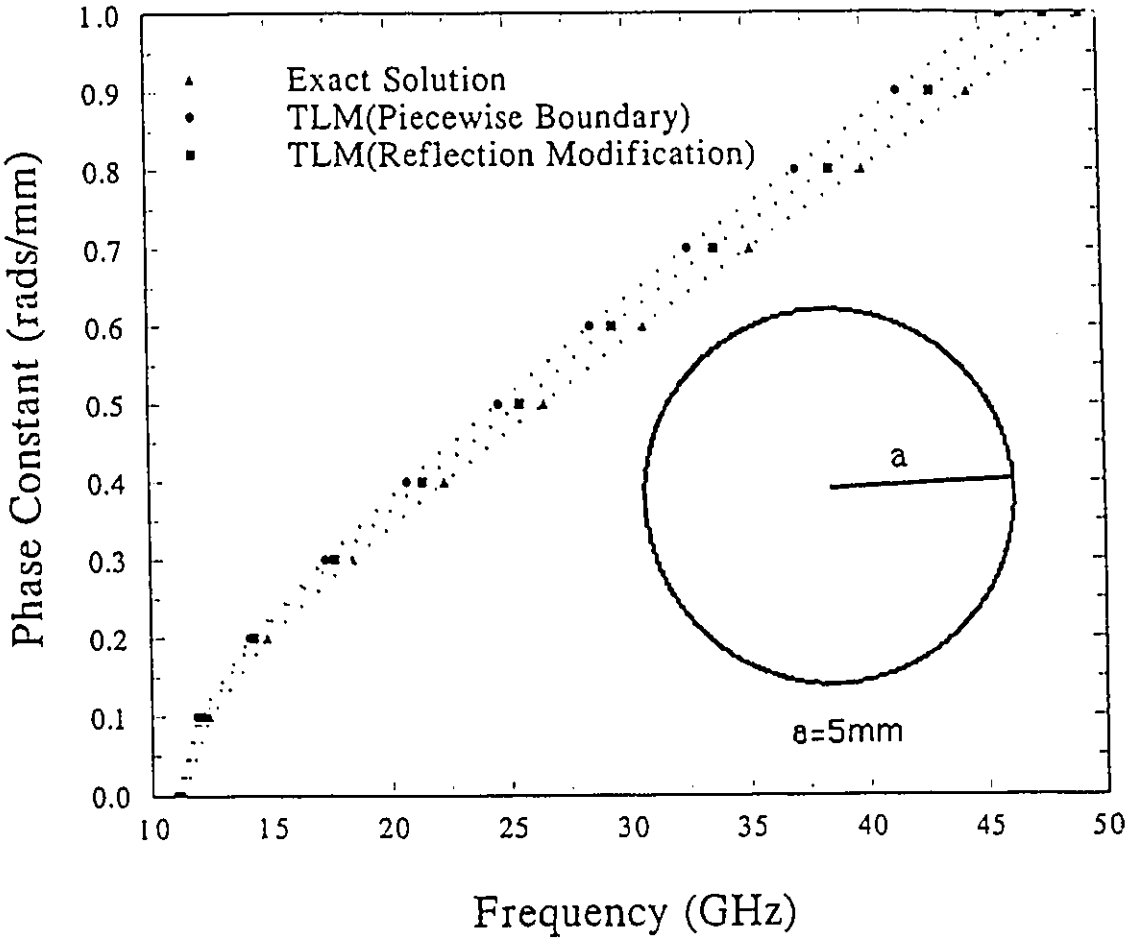

Figure 6.16: Dispersion characteristics of TE_{11} mode of empty circular waveguide(a=5mm, 8×8 mesh size)

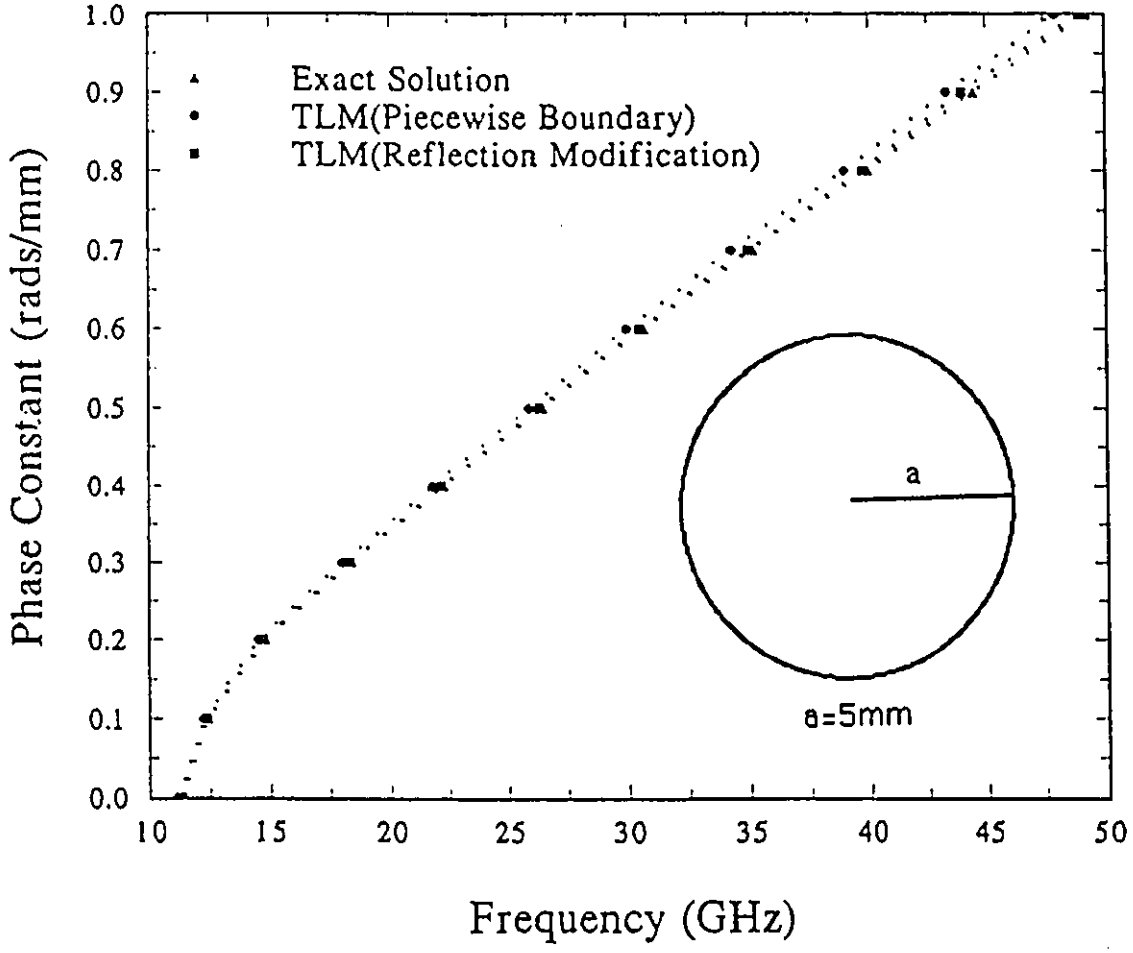

Figure 6.17: Dispersion characteristics of TE_{11} mode of empty circular waveguide(a=5mm)

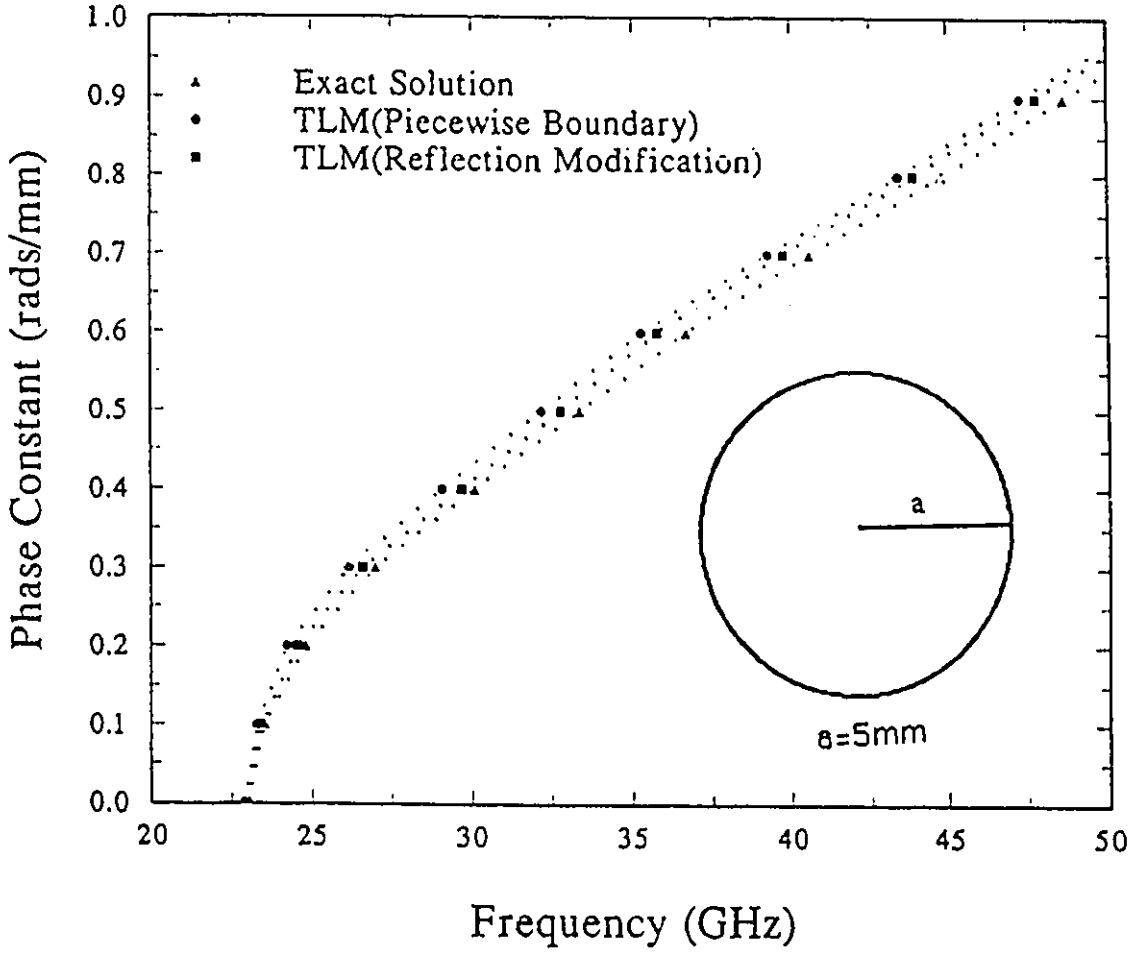

Figure 6.18: Dispersion characteristics of TM_{01} mode of empty circular waveguide(a=5mm)

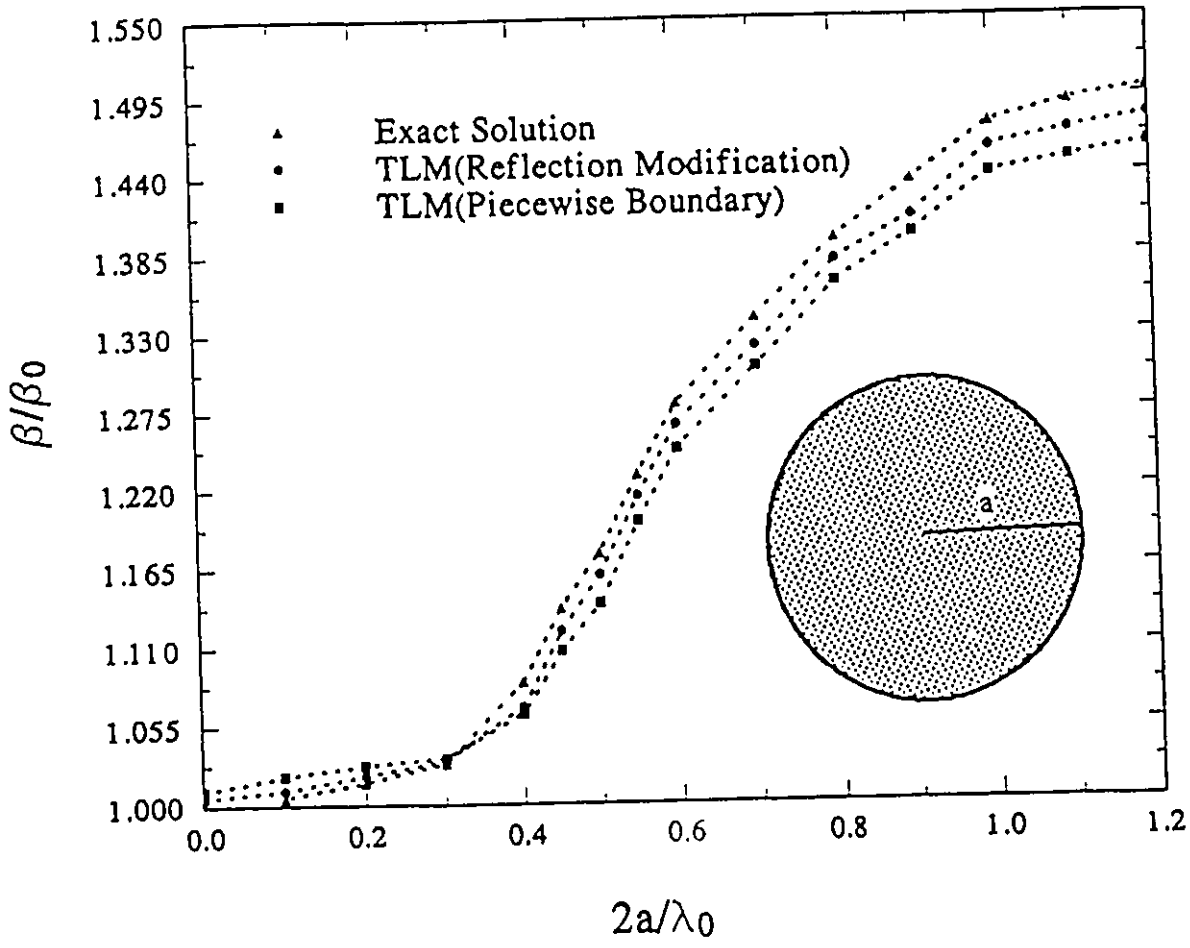

Figure 6.19: Dispersion characteristics of HE_{11} mode of dielectric circular waveguide ($\epsilon_r = 2.56$)

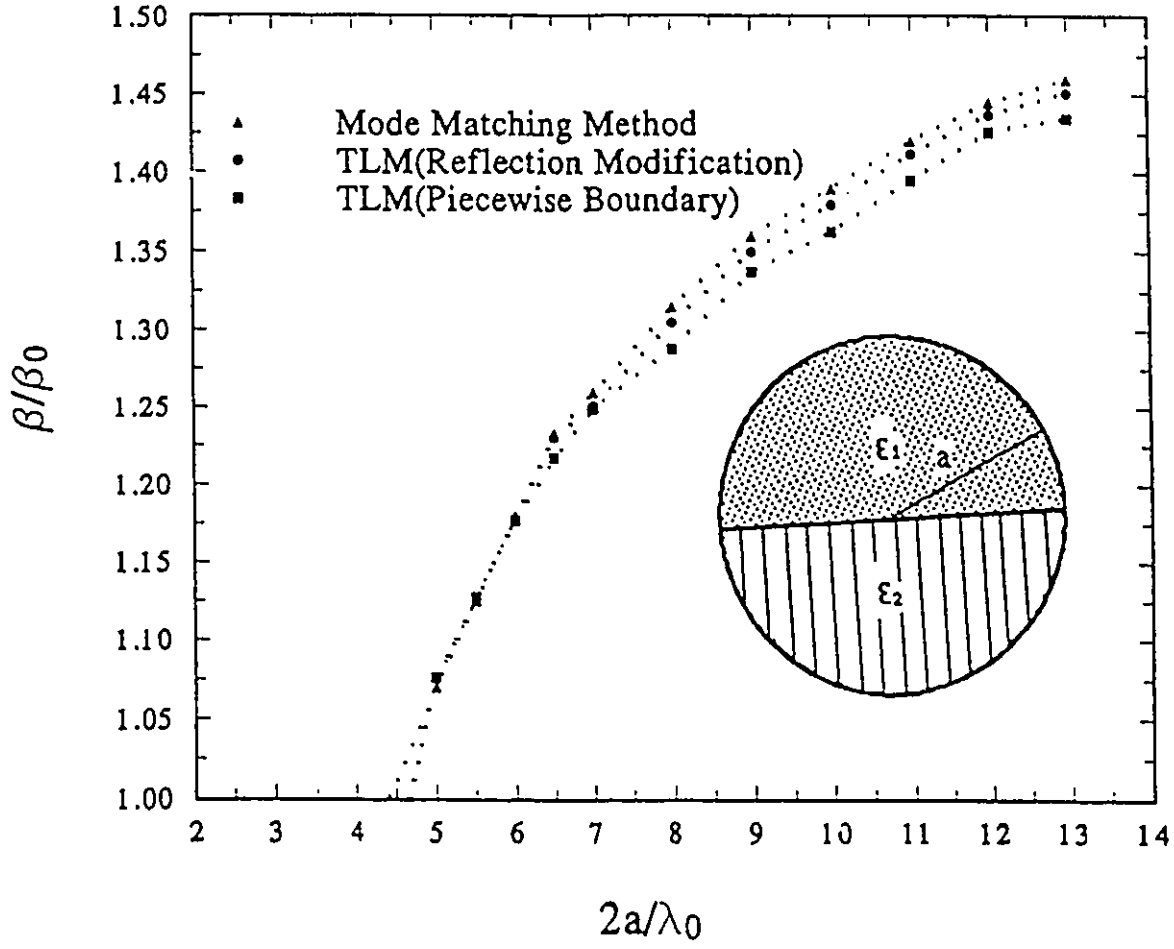

Figure 6.20: Dispersion characteristics of ${}_0TE_{01}$ mode of composite circular waveguide (ϵ_1=2.56,ϵ_2 = 2.03)

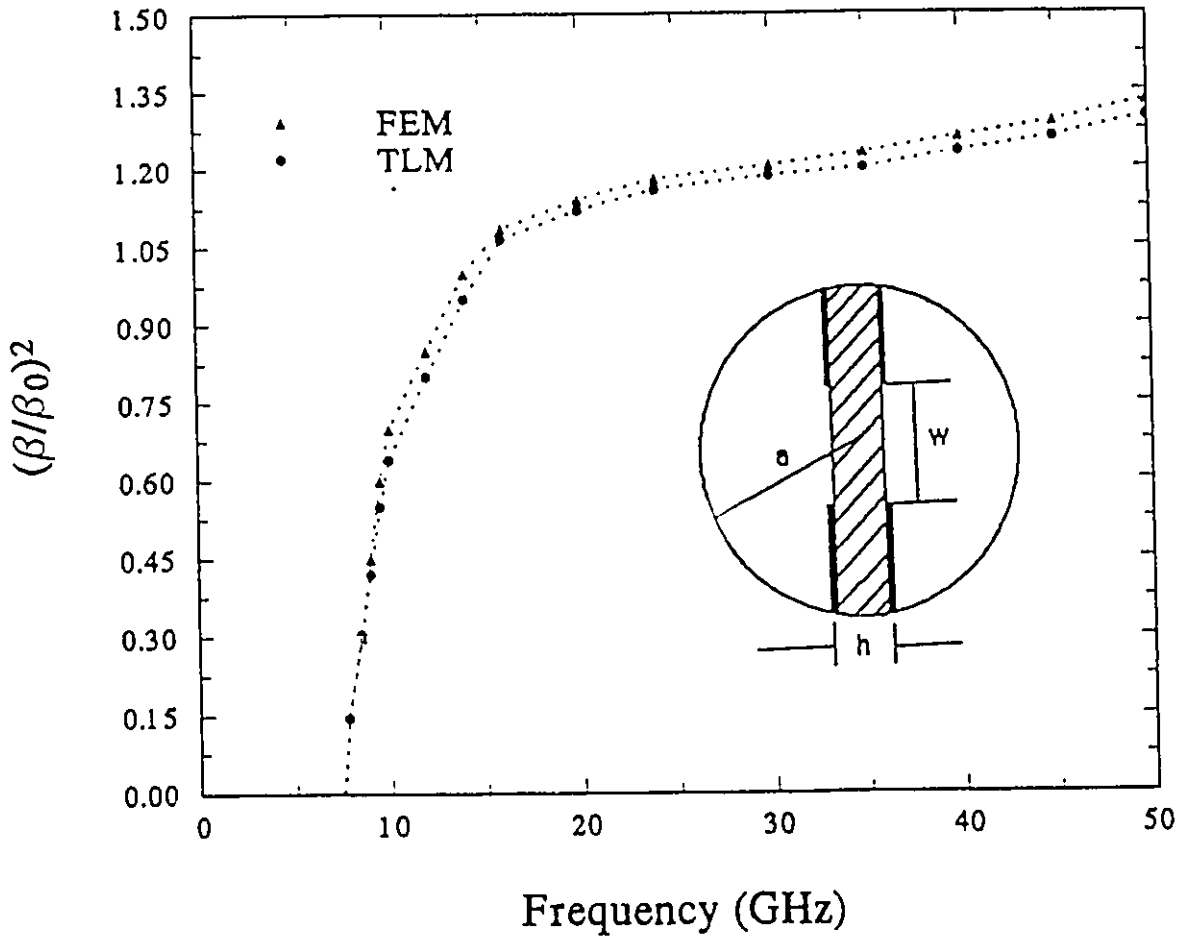

Figure 6.21: Dispersion characteristics of HE_{11} mode of a bilateral finline in circular waveguide housing(WC33) a=4.156mm, h=0.254mm, w=0.3mm, ϵ_r = 2.2)

Chapter 7

Conclusions

This thesis describes the physical principle, the formulation, the implementation of the TLM 3D symmetrical condensed node. Numerous features and applications of the method are discussed, such as the treatment of inhomogeneities, losses and anisotropic properties material.

An improved TLM algorithm which allows a full-wave analysis and computation of dispersion characteristics of guided wave structures was described. Introducing a phase shift in the longitudinal direction and choosing the propagation constant as input parameter, allows the determination of the mode corresponding frequency by using a Fourier transform of the TLM impulse response. In contrast to the conventional TLM full-wave analysis, the new approach reduces the three dimensional mesh to only a two-dimensional network. This leads to a significant reduction of CPU time and memory space requirement.

A reflection coefficient technique applied to the 2D and 3D symmetrical condensed node was introduced to deal with curved boundary. This technique removes the restriction that dimensions of structure can only be integer multiples of the TLM mesh size. Thus, it considerably improves the flexibility and simulation accuracy without significantly increasing the computer expenditure as compared with traditional piecewise boundary approximation technique. Finally, to make the method valid for a wide class of guides, absorbing

boundaries conditions was implemented for application to open guiding structures.

Dispersion characteristics of different types of rectangular and circular guided wave structures were presented. It was found that results with the proposed technique are within 5%, as compared with other methods provided that $\Delta l/\lambda < 0.05$, which is the standard requirement of the basic TLM algorithm in order to limit error due to network dispersion.

The development of this new 3D TLM symmetrical condensed node algorithm, the techniques of curved boundary treatment and absorbing boundary condition have contributed to substaintially decrease the computer cost for establishing the dispersion characteristics of waveguides. However, in the case of multimode operation in optical guides, the characterization of higher order modes become difficult because of the relatively small size of TLM mesh which is required in order to reduce dispersion error over the whole frequency range of operation. Some future work would be to take the advantage of the axi-symmetrical geometry of these structures by introducing in the TLM scheme, the azimuthal dependence. This may increase the CPU time but reduce significantly the memory requirement.

Bibliography

[1] P.B.Johns and R.L.Beurle, "Numerical Solution of 2-dimensional Scattering Problems Using A Transmission-Line Matrix", Pro. Inst. Electro. Eng., Vol.118, pp 1203-1208, Sep. 1971

[2] P.B.Johns, "Application of The Transmission-Line Matrix Method to Homogeneous Waveguides of Arbitrary Cross-section", Proc. Inst. Electr. Eng., Vol.119, pp. 1086-1091, Aug.1972

[3] P.B.Johns, "The Solution of Inhomogeneous Waveguide Problems Using a Transmission-Line Matrix", IEEE Trans. Microwave Theory Tech., Vol. MTT-22, pp.209-215, Mar. 1974

[4] S.Arhtarzard and P.B.Johns, "Numerical Solution of Lossy Waveguides: TLM Computer Program", Electron. Lett., Vol.10,pp.309-311, July 25, 1974

[5] Y.C.Shih, W.J.R.Hoefer and A.Ros, "Cutoff Frequencies In Fin Lines Calculated With A Two-dimensional TLM-program", IEEE-MTT Int. Microwave Symp. Dig., pp.261-263, June 1989

[6] Z.Zhen, M.M.Ney and W.J.R.Hoefer, "A New Boundary Description in Two-dimensional TLM Models of Microwave Circuit", IEEE Trans. on MTT, Vol.39, no.3, Mar. 1991

[7] P.Johns, "Transient Analysis of Waveguides with curved Boundaries", Electron. Lett., Vol.9, No.21, 18th Oct. 1973

[8] W.J.R.Hoefer, "The Transmission Line Matrix(TLM) Method", in Numerical Techniques For Microwave and Millimeter-wave Passive Structure(ed.T.Itoh), New York, Johns Wiley & sons,Inc., 1989

[9] S.Akhtarzad and P.B.Johns, "'Solution of 6-components Electromagnetic Fields in Three Space dimensions and Time by the TLM Method", Electron. Lett., Vol.10,pp.535-537, Dec. 12, 1974

[10] S.Akhtarzad and P.B.Johns, "TLM Analysis of Microstrip Lines on Magnetic Substrates Using 3-dimensional Resonators", Electron. Lett.,Vol.11,pp.130-131, Mar.20, 1975

[11] S.Akhtarzad and P.B.Johns, "Dispersion Characteristic of a Microstrip Line With a Step Discontinuity", Electron. Lett., Vol.11, pp.310-311, July 10, 1975

[12] S.Akhtazad, "Analysis of Lossy Microwave Structures and Microstrip Resonators by the TLM Method", Ph.D. Dissertation, University of Nottingham, England, July 1975

[13] "Analysis of Microstrip Lines on Inhomogeneous Anisotropic Substrates by the TLM Numerical Technique", Ph.D. Thesis, University of California, Los Angeles, June 1978

[14] N.Yoshida, I.Fukaiand J.Fukuoka, "Transient Analysis of Three-dimensional Electromagnetic Fields by Nodal Equations", Tran. Inst. Electron. Commun. Eng. JPN., Vol. J63B, pp.876-883, sep,1980

[15] D.A.AL-Mukhtar and J.E.Sitch, "Transmission-Line Matrix Method With Irregularly Graded Space", IEE Pro., part H: Microwave, Opt. Antennas, Vol.128, pp.229-305, Dec.1981

[16] G.E.Mariki and C.Yeh, "Dynamic Three-dimensional TLM Analysis of Microstrip Lines on Anisotropic Substrates", IEEE Trans. Microwave Theory Tech., Vol. MTT-33, pp.789-799, sep.1985

[17] W.J.R.Hoefer, "The Transmission-Line Matrix Method-Theory and Application", IEEE Trans. Microwave Theory Tech., Vol. MTT-33, pp.882-893. Oct. 1985

[18] P.B.Johns, "New symmetrical Condensed Node for Three-dimensional Solution of Electromagnetic-wave Problems by TLM", Electron. Lett.,Vol.22,pp.162-164, Jan. 1986

[19] P.B.Johns, "A symmetrical Condensed Node for TLM Method", IEEE Trans. Microwave Theory Tech., Vol. MTT-35, pp.370-377, Apr. 1987

[20] R.Allen, A.Mallik and P.B.Johns, "Numerical Results for the symmetrical Condensed TLM Node", IEEE Trans. Microwave Theory Tech., Vol. MTT-35,pp.378-382,Apr.1987

[21] J.S.Neilsen and W.J.R.Hoefer, "Modification of the Condensed 3-D TLM Node to Improve modelling of Conductor Edges", IEEE Microwave and Guided Wave Letters, Vol.2,No.3, Mar. 1992

[22] W.J.R.Hoefer, "CAD Oriented modelling of Electromagnetic Field in the Time Domain Using TLM Techniques", Workshop on " CAD oriented Numerical Techniques

for Analysis of Microwave and MM-wave Transmission Line Discontinuities and Junctions", Stuttgart, Sep.13, 1991

[23] W.J.R.Hoefor, "Time Domain Electromagnetic Simulation for Microwave CAD Applications", IEEE Trans. on Microwave and Techniques, Vol.40, No.7, July 1992

[24] Eswarappa,G.Costache and W.J.R.Hoefer, "TLM modelling of Dispersive Wideband Absorbing Boundaries With Time Domain Diakoptics for S-parameters Extraction", IEEE Trans. on Microwave Theory Tech., Vol, MTT-41, April 1990

[25] Ulf *etal*, "Simulations With 3D-TLM SCN Using FDTD Absorbing Boundary Conditions", 1992 IEET MTT-S Digest,pp377-379

[26] Chen *etal*, "Study of Absorbing Boundary Conditions in the 3D-TLM Symmetrical Condensed Node Model", 1992 IEEE MTT-S Digest, pp.369-372

[27] Z.Z.Chen, "Boundary Treatment in TLM method", ph.D thesis, University of Ottawa

[28] H.Jin,R.Vahldieck and S.Xiao, " An Improved TLM full-wave Analysis Using a Two-dimensional Mesh" IEEE MTT-s Int. Microwave Symp. Dig., Boston, MA,June 1991, pp.675-677

[29] S.Xiao,R.Vahldieck and H.Jin, "Full-wave Analysis of Guided Wave Structures Using a Novel 2-D FDTD", IEEE Microwave and Guide Wave Letters, Vol.2, No.5, May 1992

[30] A.K.Tiwari, B.Bhat and R.P.Singh, "Generalized Coupled Dielectric Waveguide and Its Variants for Millimeter-wave Applications", IEEE Trans. Microwave Tech., Vol,MTT-34, pp.869-875, Aug,1986

[31] R.E.Collin, "Field Theory of Guided Waves", Second Edition, IEEE press.

[32] R.Mittra and T.Itoh, "A New Technique of The Analysis of the Dispersion Characteristics of Microstrip Lines", IEEE Trans. Microwave Theory., Vol. MTT-19,pp.47-56. Jan.1971

[33] D.Mirshekar-syah kal and J.B.Davies, "Accurate Solution of Microstrip and Coplanar Structures for Dispersion and For Dielectric and Conductor Lossess",IEEE Trans. Microwave Theory Tech., Vol.MTT-27,pp.694-699, July 1979

[34] M.Koshiba and K.Inoue, "Simple and Efficient Finite-Element Analysis of Microwave and Optical Waveguides",IEEE Trans. on Microwave Theory and Tech. Vol.40. No.2,Feb.1992

[35] K.Hayata, K.Miura and M.Koshiba, "Finite-element formulation for lossy waveguides" IEEE Trans. on MTT, vol. 36, No.2, Feb. 1988

[36] A.K.Tiwari, B.Bhat, and R.P.Singh, "Generalized coupled dielectric waveguide and its variants for millimeter-wave application", IEEE Trans. on MTT ,vol. 34, pp.869-875, August, 1986

[37] J.E.Goell, "A circular harmonic computer analysis of rectangular dielectric waveguides" Bell sys. Tech. J., vol 48, pp.2133-2160, Sept., 1969

[38] C.Yeh, K.Ha, S.B.Dong,and W.P.Brown, "Single-mode optical waveguides", Appl. Opt., vol.18. pp.1490-1504, May 1979

[39] X.Zhang, J.Fang, K.K.Mei, "Calculations of the Dispersion characteristics of Microstrips by Time-Domain Finite Difference Method", IEEE Trans. on Microwave Theory and Techniques, Vol.36, No.2, Feb.1988

[40] P.Pramanick and P.Bhartia, "An accurate descreption of dispersion in microstrip", characteristics of Microstrips by Time-Domain Finite Difference Method", Microwave J. PP.89-96, Dec, 1983.

[41] E. Yamashita,R.Knznya, "Composite dielectric waveguides",IEEE, Trans. on MTT, Vol.28, No.9. Sep. 1980

[42] Eswarappa, G. Costache, W.R.Hoefer, "Finlines in rectangular and circular waveguide housing including substrate mounting and bending effects-finite element analysis", IEEE Trans. on Microwave Theory and Techniques, Vol.37, No.2, Feb.1989

[43] R.E.Collin, Foundations of Microwave Engineering. New York: Mcgraw Hill 1966

[44] C.D.Taylor, D.-H.Lan, and T.H.Shumpert, "Electromagnetic pulse in time varying inhomogeneous media",IEEE Trans. on Antennas Propagat.,vol. ap-17,pp. 585-589, sep.1969

[45] Z.chen, M.M.Ney and W.J.R.Hoefer, "A new finite-difference time-domain formulation and its equivalent with the TLM symmetrical condensed node", IEEE trans MTT-39, No.12, Dec. 1991

[46] P.Saguet, "TLM method for the three dimension analysis of microwave and mm-wave structures", International workshop of the German IEEE MTT/AP chapter on CAD oriented Numerical techniques for the analysis of microwave and MM=wave transmission-line discontinuities and junctions, stuttgart, Germany, Sep. 13 1991.

[47] C.R.Brewitt-Taylor and P.B.Johns, "On the construction and numerical solution of transmission-line and lumped network models of maxwell's equation", Int. J. Num. Meth. Eng,vol.15, pp. 13-30,1980

[48] W.J.R. Hoefer, "Huygens and the computer–A powerful alliance in numerical electromagnetic", proceedings of IEEE, vol.79, No.11, Oct. 1991

[49] W.J.R.Hoefer, "Time Domain Electromagnetic Simulation for Microwave CAD Applications" IEEE Trans. on MTT, vol.40, No.7, July 1992

[50] D.H.Choi and W.J.R.Hoefer: " The finite-Difference Time-Domain Method and its Application to Eigenvalue Problems" IEEE Trans. on MTT, vol.34, No.12,Dec., 1986.

```
/*****************************************/
/* This is the programm written in c language */
/* to calculate the dispersion characteristics of */
/* the microwave/optical guided wave structures */
/* by using transmission line matrix method */
/*****************************************/

#include <stdio.h>
#include <math.h>
#include <stddef.h>
#include <stdarg.h>
#include <complex.h>
#include <stdlib.h>
#include <string.h>
#include <sym.h>
#include <lex.h>
#include <cint.h>

#define true 1
#define false 0
#define maximum 100
#define mimimum 50

/*****************/
/* golable variables */
/*****************/
main()

{int t, i, j ,k, m, imax, jmax, tmax, ic1, jc1, ic2, jc2;
float T1, T2, T3, T4, T5, T6;

/*************************************************/
/* ic1 and jc1 kc1 are excitation points. ic2, jc2, kc2 */
/* are extraction points. tmax is the maximum number */
/* of iterations. imax and jmax kmax are maximun */
/* values of caculation box. T1,T2,T3,T4,T5,T6 are the */
/* reflection coefficients of each boundary. */
/*************************************************/

float medium1, medium2, time_total1[4000],
float time_total2[4000], sub1, sub2;
float sum1, sum2, sumsin, sumcos, sum;
float y, z, y0, z0, g0, r0, er, ur, thegma,
float a1, b1, c1, d1, e1, f1, g1, h1, i1, j1;

float u[18][30][30], v[18][30][30];

/*************************************************/
/* u is reflection wave, v is incident wave. time-total is */
/* the time domain er and ur are permittibity and */
/* premeability respectively. thegma is the conductivity*/
/*************************************************/

float f, fmax, deltal, a,b, l_w, belta;
```

```
FILE *output1, *output2, *output3;

/*****************************************************/
/* fmax is the maxmium frequency in frequency domain,          */
/* deltal is the mesh size, a,b are length*width respectively  */
/* in mm belta is the propagation factor. l_w is the mesh      */
/* size divided by wavelength                                  */
/*****************************************************/

tmax=500;
ic1=15;
jc1=15;
ic2=14;
jc2=16
T1=T2=T3=T4=T5=T6=-1.0;

belta=1.2;
fmax=25.0;
a=12.7;
b=12.7;
er=8.875;
ur=2.0;
thegma=0.0;
r0=0.0;

output1=fopen("tlmt2.dat","w");

/*************************************/
/* imax and jmax will be decided according to  */
/*     the following procedures                */
/*************************************/

deltal=300.0*0.05/(1.414*fmax);

imax=a/deltal;
jmax=b/deltal;
y0=2.0*(er-2.0);
z0=2.0*(ur-2.0);
g0=thegma*deltal*377.0;

/*********************************************/
/* values are given to a1, b1, c1, d1, e1, f1, g1, h1, i1,j1*/
/*********************************************/

printf("imax=%d,jmax=%d\n",imax,jmax);

/***********************/
/*        Initial Values        */
/***********************/

for (i=0;i<=imax;++i)
{
for (j=0;j<=jmax;++j)
{
```

```
for (m=0;m<18;++m)
{
v[m][i][j]=0.0;
}
}
}

/* The value of excitation point */

v[2][ic1][jc1]=1.0/2.0;
v[3][ic1][jc1]=1.0/2.0;
v[7][ic1][jc1]=1.0/2.0;
v[10][ic1][jc1]=1.0/2.0;
v[13][ic1][jc1]=1.0/2.0;

/************************/
/*       The Algorithm          */
/************************/

for(k=1;k<=2;++k)
{
if (k==1)
{medium1=cos(belta*deltal);
medium2=cos(belta*deltal);
}
else
{   medium1=sin((-belta)*deltal);
medium2=sin(belta*deltal);
for (i=0;i<=imax;++i)
{
for (j=0;j<=jmax;++j)
{
for (m=0;m<18;++m)
{
v[m][i][j]=0.0;
u[m][i][j]=0.0;
}
}
}
/***************************/
/* The value of excitation point */
/***************************/
v[2][ic1][jc1]=1.0/2.0;
v[3][ic1][jc1]=1.0/2.0;
v[7][ic1][jc1]=1.0/2.0;
v[10][ic1][jc1]=1.0/2.0;
v[13][ic1][jc1]=1.0/2.0;
}
for (t=1;t<=tmax;++t)
{
for (i=0;i<=imax;++i)
{
for (j=0;j<=jmax;++j)
{
```

```
if(((i>=2)&&(i<=9))&&((j>=2)&&(j<=5)))
if((j>=0)&&(j<=2))
{
a1=(-y0)/(2.0*(y0+4.0))+(z0)/(2.0*(4.0+z0));
b1=4.0/(2.0*(4.0+y0));
c1=-(y0)/(2.0*(4.0+y0))-(z0)/(2.0*(4.0+z0));
d1=4.0/(2.0*(4.0+z0));
e1=b1;
f1=z0*d1;
g1=y0*b1;
h1=(y0-4.0)/(y0+4.0);
i1=d1;
j1=(4.0-z0)/(4.0+z0);
}
if (((i>=14)&&(i<=21))&&((j>=2)&&(j<=5)))
{
a1=(-y0)/(2.0*(y0+4.0))+(z0)/(2.0*(4.0+z0));
b1=4.0/(2.0*(4.0+y0));
c1=-(y0)/(2.0*(4.0+y0))-(z0)/(2.0*(4.0+z0));
d1=4.0/(2.0*(4.0+z0));
e1=b1;
f1=z0*d1;
g1=y0*b1;
h1=(y0-4.0)/(y0+4.0);
i1=d1;
j1=(4.0-z0)/(4.0+z0);
}
else
{
a1=0.0;
b1=1.0/2.0;
c1=0.0;
d1=1.0/2.0;
e1=1.0/2.0;
f1=0.0;
g1=0.0;
h1=-1.0;
i1=1.0/2.0;
j1=1.0;
}

/*************************/
/* Substitute the next values    */
/*************************/

u[0][i][j]=a1*v[0][i][j]+b1*v[1][i][j]+d1*v[2][i][j]+b1*v[8][i][j]
-d1*v[10][i][j]+c1*v[11][i][j]+g1*v[12][i][j]+i1*v[17][i][j];

u[1][i][j]=b1*v[0][i][j]+a1*v[1][i][j]+d1*v[5][i][j]+c1*v[8][i][j]
-d1*v[9][i][j]+b1*v[11][i][j]+g1*v[12][i][j]+i1*v[16][i][j];

u[2][i][j]=d1*v[0][i][j]+a1*v[2][i][j]+b1*v[3][i][j]+b1*v[7][i][j]
+c1*v[10][i][j]-d1*v[11][i][j]+g1*v[13][i][j]-i1*v[17][i][j];
```

```
u[3][i][j]=b1*v[2][i][j]+a1*v[3][i][j]+d1*v[4][i][j]-d1*v[6][i][j]
+c1*v[7][i][j]+b1*v[10][i][j]+g1*v[13][i][j]+i1*v[15][i][j];

u[4][i][j]=d1*v[3][i][j]+a1*v[4][i][j]+b1*v[5][i][j]+c1*v[6][i][j]
-d1*v[7][i][j]+b1*v[9][i][j]+g1*v[14][i][j]-i1*v[15][i][j];

u[5][i][j]=d1*v[1][i][j]+b1*v[4][i][j]+a1*v[5][i][j]+b1*v[6][i][j]
-d1*v[8][i][j]+c1*v[9][i][j]+g1*v[14][i][j]+i1*v[16][i][j];

u[6][i][j]=-d1*v[3][i][j]+c1*v[4][i][j]+b1*v[5][i][j]+a1*v[6][i][j]
+d1*v[7][i][j]+b1*v[9][i][j]+g1*v[14][i][j]+i1*v[15][i][j];

u[7][i][j]=b1*v[2][i][j]+c1*v[3][i][j]-d1*v[4][i][j]+d1*v[6][i][j]
+a1*v[7][i][j]+b1*v[10][i][j]+g1*v[13][i][j]-i1*v[15][i][j];

u[8][i][j]=b1*v[0][i][j]+c1*v[1][i][j]-d1*v[5][i][j]+a1*v[8][i][j]
+d1*v[9][i][j]+b1*v[11][i][j]+g1*v[12][i][j]+i1*v[16][i][j];

u[9][i][j]=-d1*v[1][i][j]+b1*v[4][i][j]+c1*v[5][i][j]+b1*v[6][i][j]
+d1*v[8][i][j]+a1*v[9][i][j]+g1*v[14][i][j]-i1*v[16][i][j];

u[10][i][j]=-d1*v[0][i][j]+c1*v[2][i][j]+b1*v[3][i][j]+b1*v[7][i][j]
+a1*v[10][i][j]+d1*v[11][i][j]+g1*v[13][i][j]+i1*v[17][i][j];

u[11][i][j]=c1*v[0][i][j]+b1*v[1][i][j]-d1*v[2][i][j]+b1*v[8][i][j]
+d1*v[10][i][j]+a1*v[11][i][j]+g1*v[12][i][j]-i1*v[17][i][j];

u[12][i][j]=c1*v[0][i][j]+c1*v[1][i][j]+c1*v[8][i][j]+c1*v[11][i][j]
+h1*v[12][i][j];

u[13][i][j]=c1*v[2][i][j]+c1*v[3][i][j]+c1*v[7][i][j]+c1*v[10][i][j]
+h1*v[13][i][j];

u[14][i][j]=c1*v[4][i][j]+c1*v[5][i][j]+c1*v[6][i][j]+c1*v[9][i][j]
+h1*v[14][i][j];

u[15][i][j]=f1*v[3][i][j]-f1*v[4][i][j]+f1*v[6][i][j]-f1*v[7][i][j]
+j1*v[15][i][j];

u[16][i][j]=-f1*v[1][i][j]+f1*v[5][i][j]+f1*v[8][i][j]-f1*v[9][i][j]
+j1*v[16][i][j];

u[17][i][j]=f1*v[0][i][j]-f1*v[2][i][j]+f1*v[10][i][j]-f1*v[11][i][j]
+j1*v[17][i][j];

}
}

/*****************************/
/*        Boundary Conditions        */
/*****************************/

for (i=0;i<=imax;++i)
{
```

```
}

if((i==imax)&&((j>0)&&(j<jmax)))
{
v[9][i][j]=T6*u[9][i][j];
v[10][i][j]=T6*u[10][i][j];
v[11][i][j-1]=u[0][i][j];
v[6][i][j-1]=u[4][i][j];
v[0][i][j+1]=u[11][i][j];
v[6][i][j+1]=u[6][i][j];
v[i+1][j]=u[5][i][j];
v[10][i-1][j]=u[2][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
}
if((i>=13)&&(i<=15)&&(j==3))
{
v[11][i][j]=-u[11][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[10][i-1][j]=u[2][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[6][i][j]=-u[6][i][j];
v[9][i-1][j]=u[5][i][j];
v[4][i][j+1]=u[6][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[0][i][j+1]=u[11][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
}

if((i>=13)&&(i<=15)&&(j==4))
{
v[11][i][j-1]=u[0][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[10][i-1][j]=u[2][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[6][i][j-1]=u[4][i][j];
```

```
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
}

if(((i>0)&&(i<imax))&&(j==jmax))
{
v[11][i][j]=T6*u[11][i][j];
v[6][i][j]=T6*u[6][i][j];
v[11][i][j-1]=u[0][i][j];
v[6][i][j-1]=u[4][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[9][i-1][j]=u[5][i][j];
v[10][i-1][j]=u[2][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
}

if((i==imax)&&(j==jmax))
{
v[9][i][j]=T6*u[9][i][j];
v[10][i][j]=T6*u[10][i][j];
v[11][i][j]=T5*u[11][i][j];
v[6][i][j]=T5*u[6][i][j];
v[11][i][j-1]=u[0][i][j];
v[6][i][j-1]=u[4][i][j];
v[9][i-1][j]=u[5][i][j];
v[10][i-1][j]=u[2][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
```

```
v[5][i][j]=T3*u[5][i][j];
v[4][i][j+1]=u[6][i][j];
v[0][i][j+1]=u[11][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
}

if((i==0)&&((j<0)&&(j<jmax)))
{
v[2][i][j]=T3*u[2][i][j];
v[5][i][j]=T3*u[5][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[0][i][j+1]=u[11][i][j];
v[6][i][j+1]=u[6][i][j];
v[11][i][j-1]=u[0][i][j];
v[6][i][j-1]=u[4][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
}

if((i==0)&&(j==jmax))
{
v[2][i][j]=T3*u[2][i][j];
v[5][i][j]=T3*u[5][i][j];
v[11][i][j]=T5*u[11][i][j];
v[6][i][j]=T5*u[6][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[11][i][j-1]=u[0][i][j];
v[6][i][j-1]=u[4][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
```

```
for (j=0;j<=jmax;++j)
{

if ((i==imax)&&(j==0))
{
v[0][i][j]=T1*u[0][i][j];
v[4][i][j]=T1*u[4][i][j];
v[9][i][j]=T6*u[9][i][j];
v[10][i][j]=T6*u[10][i][j];
v[10][i-1][j]=u[2][i][j];
v[9][i-1][j]=u[5][i][j];
v[0][i][j+1]=u[6][i][j];
v[4][i][j+1]=u[11][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto nextj;
}

if((j==0)&&((i>0)&&(i<imax)))
{
v[0][i][j]=T1*u[0][i][j];
v[4][i][j]=T1*u[4][i][j];
v[9][i-1][j]=u[5][i][j];
v[10][i-1][j]=u[2][i][j];
v[4][i][j+1]=u[6][i][j];
v[0][i][j+1]=u[11][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto nextj;
}

if((i==0)&&(j==0))
{
v[0][i][j]=T1*u[0][i][j];
v[4][i][j]=T1*u[4][i][j];
v[2][i][j]=T3*u[2][i][j];
```

```
v[9][i-1][j]=u[5][i][j];
v[4][i][j]=-u[4][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[0][i][j]=-u[0][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];
goto  nextj;
}

v[11][i][j-1]=u[0][i][j];
v[8][i][j]=medium1*u[1][i][j];
v[10][i-1][j]=u[2][i][j];
v[7][i][j]=medium1*u[3][i][j];
v[6][i][j-1]=u[4][i][j];
v[9][i-1][j]=u[5][i][j];
v[4][i][j+1]=u[6][i][j];
v[3][i][j]=medium2*u[7][i][j];
v[1][i][j]=medium2*u[8][i][j];
v[5][i+1][j]=u[9][i][j];
v[2][i+1][j]=u[10][i][j];
v[0][i][j+1]=u[11][i][j];
v[12][i][j]=u[12][i][j];
v[13][i][j]=u[13][i][j];
v[14][i][j]=u[14][i][j];
v[15][i][j]=-u[15][i][j];
v[16][i][j]=-u[16][i][j];
v[17][i][j]=-u[17][i][j];

nextj:;
}
}
if(k==1)
{
time_total1[t]=2.0*(v[2][ic2][jc2]+v[3][ic2][jc2]
+v[7][ic2][jc2]+v[10][ic2][jc2]+y0*v[13][ic2][jc2])/4.0;
}
else
{
/*   printf("k=%d\n",k);   */
time_total2[t]=2.0*(v[2][ic2][jc2]+v[3][ic2][jc2]
+v[7[[ic2][ij2]+v[10][ic2][jc2]+y0*v[13][ic2][jc2])/4.0;
/*   printf("time=%f\n",time_total2[t]);   */
}
}  /* t is finished */
}  /* k is finished */

for(l_w=0.0008;l_w<=0.05;l_w=l_w+0.0002)
```

```
{
sum1=0.0;
sum2=0.0;
for(t=1;t<=tmax;++t)
{
sub1=time_total1[t]*cos(2*3.1415926*t*l_w);
sub2=time_total1[t]*sin(2*3.1415926*t*l_w);
sum1=sub1+sum1;
sum2=sub2+sum2;
}
sumcos=sqrt(sum1*sum1+sum2*sum2);
/*  printf(" sum1=%f\n", sum1);  */
sum1=0.0;
sum2=0.0;
/*    printf(" sum1=%f\n", sum1);  */
for(t=1;t<=tmax;++t)
{
sub1=time_total2[t]*cos(2*3.1415926*t*l_w);
sub2=time_total2[t]*sin(2*3.1415926*t*l_w);
sum1=sub1+sum1;
sum2=sub2+sum2;
}
sumsin=sqrt(sum1*sum1+sum2*sum2);
sum=sqrt(sumsin*sumsin+sumcos*sumcos);

f=2.0*l_w*300.0/(deltal);   /* f is in GHz */
printf("f=%f,  sum=%f\n",  f,  sum);

/*  fprintf(output1,"f=%f,  sum=%f\n",  f,  sum);  */

}
}
```

```
/*********************************************/
/* This is the program to calculate the cutoff frequency */
/*       of waveguide by using 2D TLM method                */
/*********************************************/

#include <stdio.h>
#include <math.h>

main()
{ int t, i, j, m, imax, jmax, tmax, ic1, jc1, ic2, jc2;
float T1, T2, T3, T4;

/*************************************************************/
/* ic1 and jc1 are excitation points. ic2 and jc2 re extraction points.   */
/* tmax is the maximum number of iterations. imax and jmax are            */
/* maximun values of caculation box. T1,T2,T3,T4 are the reflection       */
/* coefficients of each boundary. l_w is mesh size divided by wavelength */
/*************************************************************/

float time_total[3501], sub1, sub2, sum1, sum2, sum, frquncy_total;
float u[4][100][60], v[4][100][60];

/*************************************************************/
/* u is reflection wave, v is incident wave. time-total is the time domain*/
/* frquncy-total is frequency domain solution.                            */
/*************************************************************/

float f, fmax, deltal, a, b, l_w;
FILE *output1;

/*********************************************/
/* fmax is the maxmium frequency in frequency domain, deltal is  */
/* the mesh size, a and b are length*width respectively in mm     */
/*********************************************/

tmax=3500;
ic1=8; /* the excitation point must be on the maximun node*/
jc1=39;
ic2=88;
jc2=4;
T1=T2=T3=T4=1.0;

fmax=43.0; /* in GHz */
a=22.8;    /* in mm */
b=10.15;   /* in mm */

output1=fopen("tlmt1.dat","w");
/*************************************/
/* imax and jmax will be decided according to  */
/*    the following procedures                 */
/*************************************/

deltal=300.0*0.05/(1.414*fmax);
/*deltal is in mm, fmax is in GHz*/
```

```
imax=a/deltal+1.0;
jmax=b/deltal+1.0;
printf("\nimax=%d,jmax=%d\n",imax,jmax);

/*************************/
/*        Initial Values          */
/*************************/

/* All the initial values are zero */

for (i=0;i<=imax;++i)
{
for (j=0;j<=jmax;++j)
{
for (m=0;m<=3;++m)
{
v[m][i][j]=0.0;
}
}
}

/* The value of excitation point is one */

v[2][ic1][jc1]=1.0;

/***********************/
/*       The Algoritm          */
/***********************/

for (t=1;t<=tmax;++t)
{
for (i=0;i<=imax;++i)
{
for (j=0;j<=jmax;++)
{
/* Substitute the next values */

/* B */ u[2][i][j]=1.0/2.0*(-v[2][i][j]+v[0][i][j]
                            +v[1][i][j]+v[3][i][j])
/* C */ u[3][i][j]=1.0/2.0*(-v[3][i][j]+v[2][i][j]
                            +v[0][i][j]+v[1][i][j]);
/* D */ u[0][i][j]=1.0/2.0*(-v[0][i][j]+v[1][i][j]
                            +v[2][i][j]+v[3][i][j]);
/* E */ u[1][i][j]=1.0/2.0*(-v[1][i][j]+v[0][i][j]
                            +v[2][i][j]+v[3][i][j]);
}
}
/*****************************/
/*             Boundary Conditions       */
/*****************************
/* Boundary 1 */
for (i=0;i<=imax;++i)
{
```

```
for (j=0;j<=jmax;++)
{
if ((i==imax)&&(j==0))
{
v[0][i][j]=T1*u[0][i][j];
v[3][i][j]=T3*u[3][i][j];
v[3][i-1][j]=u[1][i][j];
v[0][i][j+1]=u[2][i][j];
goto nextj;
}
if((j==0)&&((i>0)&&(i<imax)))
{
v[0][i][j]=T1*u[0][i][j];
v[3][i-1][j]=u[1][i][j];
v[1][i+1][j]=u[3][i][j];
goto nextj;
}
/* Boundary 2 */

if ((i==0)&&((j>0)&&(j<jmax))
{
v[1][i][j]=T2*u[1][i][j];
v[2][i][j-1]=u[0][i][j];
v[0][i][j+1]=u[2][i][j];
v[1][i+1][j]=u[3][i][j];
goto nextj;
}
if ((i==0)&&(j==0))
{
v[1][i][j]=T2*u[1][i][j];
v[0][i][j]=T2*u[0][i][j];
v[0][i][j+1]=u[2][i][j];
v[1][i+1][j]=u[3][i][j];
goto nextj;
}
/* Boundary 3 */

if (((i>0)&&(i<imax))&&(j==jmax))
 {
v[2][i][j]=T3*u[2][i][j];
v[2][i][j-1]=u[0][i][j];
v[3][i-1][j]=u[1][i][j];
v[1][i+1][j]=u[3][i][j];
goto nextj
}
if ((i==0)&&(j==jmax))
{
v[2][i][j]=T3*u[2][i][j];
v[1][i][j]=T3*u[1][i][j];
v[2][i][j-1]=u[0][i][j];
v[1][i+1][j]=u[3][i][j];
goto nextj;
}
```

```
 /* Boundary 4 */

if ((i==imax)&&((j>0)&&(j<jmax)))
 {
v[3][i][j]=T4*u[3][i][j];
v[2][i][j-1]=u[0][i][j];
v[3][i-1][j]=u[1][i][j];
v[0][i][j+1]=u[2][i][j];
goto nextj;;
}
if ((i==imax)&&(j==jmax))
{
v[3][i][j]=T4*u[3][i][j];
v[2][i][j]=T4*u[2][i][j];
v[2][i][j-1]=u[0][i][j];
v[3][i-1][j]=u[1][i][j];
goto nextj;
}

/*B*/  v[2][i][j-1]=u[0][i][j];
/*C*/  v[3][i-1][j]=u[1][i][j];
/*D*/  v[0][i][j+1]=u[2][i][j];
/*E*/  v[1][i+1][j]=u[3][i][j];
nextj:;
}
}

time_total[t]=1.0/2.0*(v[0][ic2][jc2]+v[1][ic2][jc2]+v[2][ic2][jc2]
                       +v[3][ic2][jc2]);
}/* t is finished */
for(l_w=0.0001;l_w<=0.015;l_w=l_w+0.0001)
{
sum1=0.0;
sum2=0.0;
for (t=1;t<=tmax;++t)
{
sub1=time_total[t]*cos(2.0*3.1415926*t*l_w);
sub2=time_total[t]*sin(2.0*3.1415926*t*l_w);
sum1=sub1+sum1;
sum2=sub2+sum2;
}
sum=sqrt(sum1*sum1+sum2*sum2);

 f=1.414213*l_w*300.0/deltat;  /* f is in GHz */
printf("f=%f,  sum=%f\n",  f,  sum)
fprintf(output1,"f=%f,  sum=%f\n",  f,  sum);
}
}
```

```
/************************************/
/*  This is a program to produce the graphs  */
/************************************/

/* X include files */
#include <X11/Xlib.h>
#include <X11/Xutil.h>
#include<stdio.h>
#include<string.h>
#include<math.h>
#include<ctype.h>
#define M    1000          /* frequency sample No.*/
#define ub   75           /* upper boundary          */
#define bb   600          /*bottom boundary          */
#define rb   900          /* right boundary          */
#define lb   125          /* left boundary           */

/*declarations  */
char hello[] ={"Ploter"};
char ut[80]={""};
char bt[80] ={""};
char slt[80] ={""};
char sbt[80] ={""};
FILE *fp1,*fp2;

main(argc,argv)
int  argc;
char  **argv;
{
/*declarations  */
Display   *mydisplay;
Window    mywindow;
GC mygc;
Font pointfont;
XFontStruct  *titlefontstruct,*figurefontstruct;
XEvent myevent;
KeySym mykey;
XSizeHints myhint;
int myscreen;
unsigned long myforeground, mybackground;
int i,dec,sign;
char text[10];
char header[80];
int done,x1,y1,x2,y2;
int j,k,iflag;
int  *ffx,*ivector(),*ffy,ctr,retr;
float  *rfx,*rfy;
float  xmax,xmin,ymax,ymin;
float  sxmax,sxmin,symax,symin;
float  rxmax,rxmin,rymax,rymin;
float   *fx,*fy,*tfx,*tfy;
char cfx[10][10],cfy[10][10];
float    *vector(),**cmatrix();
void  nrerror();
```

```
rfx=vector(1,M);
rfy=vector(1,M);
/*read  data*/
fp1=fopen(*(argv+1),"r");
rctr=1;
while(fscanf(fp1,"%f  %f\n",&rfx[rctr],
&rfy[rctr])!=EOF){
if(rctr==1)  rymax=rymin=rfy[rctr];
if(rymax<rfy[rctr])  rymax=rfy[rctr];
if(rymin>rfy[rctr])  rymin=rfy[rctr];
rctr=rctr+1;
}
fclose(fp1);
if(rctr<=1) nrerror("Your data file is not ready");
rctr=rctr-1;
rxmin=rfx[1];
rxmax=rfx[rctr];

/*read  titles*/
fp2=fopen("plo.title","r");
do{
if(fgets(ut,80,fp2)==NULL)  break;
if(fgets(slt,80,fp2)==NULL)  break;
if(fgets(bt,80,fp2)==NULL)  break;
if(fgets(sbt,80,fp2)==NULL)  break;
iflag= -1;
}while(iflag!=-1);

if(fscanf(fp2,"%f  %f  %f  %f\n",&sxmin,&sxmax,
&symin,&symax)==EOF) {
xmin=rxmin;
xmax=rxmax;
ymin=rymin;
ymax=rymax;
}
else {
xmin=sxmin;
xmax=sxmax;
ymin=symin;
ymax=symax;
}
fclose(fp2);

ctr=1;
fx=vector(1,rctr);
fy=vector(1,rctr);
for(k=1;k<=rctr;k++){
if(xmin<=rfx[k]&&xmax>=rfx[k]) {
fx[ctr]=rfx[k];
fy[ctr]=rfy[k];
ctr=ctr+1;
}
}
```

```
ctr=ctr-1;
/*setup  array*/
tfx=vector(1,ctr);
tfy=vector(1,ctr);
ffx=ivector(1,ctr);
ffy=ivector(1,ctr);

/* arrange the data according to the scale*/
for(k=1;k<=ctr;k++) {
ffx[k]=775.*(fx[k]-xmin)/(xmax-xmin)+125.;
}
for(k=1;k<=ctr;k++) {
ffy[k]=525.*(fy[k]-ymax)/(ymin-ymax)+75.;
}

/* set up the x-y scale*/
 for(k=0;k<=5;k++) {
  tfx[k]=(xmax-xmin)*k/5.+xmin;
  tfy[k] = (ymin-ymax)*k/5.+ymax;
  sprintf(cfx[k],"%10.2e",tfx[k]);
  sprintf(cfy[k],"%10.2e",tfy[k]);
  }

/* initialization */
mydisplay=XOpenDisplay("");
myscreen  =DefaultScreen(mydisplay);
pointfont=XLoadFont(mydisplay,"6x10");
titlefontstruct=XLoadQueryFont(mydisplay,"vr-20");
figurefontstruct=XLoadQueryFont(mydisplay,
                  "9x15");

/*default pixel values */
mybackground = WhitePixel (mydisplay, myscreen);
myforeground = BlackPixel (mydisplay, myscreen);

/* default program-specified window position and
size */
myhint.x=1;
myhint.y=1;
myhint.width=1000;
myhint.height=700;
myhint.flags= PPosition |PSize;
/* window creation  */
mywindow =XCreateSimpleWindow (mydisplay,
DefaultRootWindow (mydisplay),
myhint.x, myhint.y,myhint.width, myhint.height,5,
myforeground, mybackground);
XSetStandardProperties (mydisplay, mywindow,
hello, hello, None, argv, argc, &myhint);

/*  GC creation and initialization */
mygc =XCreateGC ( mydisplay, mywindow,0,0);
XSetBackground ( mydisplay, mygc,
mybackground);
```

```
XSetForeground (mydisplay,mygc, myforeground);

/* input event selection */
XSelectInput ( mydisplay, mywindow,
ButtonPressMask | KeyPressMask | ExposureMask);

/* window mapping */
XMapRaised (mydisplay, mywindow);

/* main event-reading loop */
done=0;
while (done==0){
 /* read the nest event */
XNextEvent (mydisplay, &myevent);
switch (myevent.type){

/* repaint window on expose events */
case Expose:
XSetLineAttributes(mydisplay,mygc,2,LineSolid,
CapRound,JoinRound);
XSetFont (mydisplay,mygc,pointfont);
for (k=1;k<ctr;k++){
if((ffy[k]>bb && ffy[k+1]>bb) ||
(ffy[k]<ub && ffy[k+1]<ub)){
if(argc<3) continue;
if(strcmp(argv[2],"-b")!=0) continue;
}
x1=ffx[k];
y1=ffy[k];
x2=ffx[k+1];
y2=ffy[k+1];
if(argc>=3){
if(strcmp(argv[2],"-b")==0) {
x2=x1;
y2=0;
}
}
if(y2>bb){
x2=x2-(x2-x1)*(y2-bb)/(y2-y1);
y2=bb;
}
if(y1>bb){
x1=x1+(x2-x1)*(y1-bb)/(y1-y2);
y1=bb;
}
if(y2<ub){
x2=x2-(x2-x1)*(y2-ub)/(y2-y1);
y2=ub;
}
if(y1<ub){
x1=x1+(x2-x1)*(y1-ub)/(y1-y2);
y1=ub;
}
XDrawLine(myevent.xexpose.display,
```

```
myevent.xexpose.window,  mygc,
x1,y1,x2,y2);
if(ffy[k]>bb || ffy[k]<ub ) continue;
if(argc>=3){
if(strcmp(argv[2],"-b")==0)  continue;
}
XDrawImageString(myevent.xexpose.display,
myevent.xexpose.window,  mygc,ffx[k]-3,
ffy[k]+2,"o",1);
}

/*write the scale of X-y axils*/
XSetFont  (mydisplay,mygc,
figurefontstruct->fid);
for(k=0;k<=5;k++){
XDrawImageString(myevent.xexpose.display,
myevent.xexpose.window,  mygc,
(rb-lb)*k/5+lb-55,bb+20,cfx[k],10);
XDrawImageString(myevent.xexpose.display,
myevent.xexpose.window,  mygc,lb-90,
(bb-ub)*k/5+ub+5,cfy[k],10);
XDrawLine(myevent.xexpose.display,
myevent.xexpose.window,  mygc,
(rb-lb)*k/5+lb,bb,(rb-lb)*k/5+lb,bb-5);
XDrawLine(myevent.xexpose.display,
myevent.xexpose.window,  mygc,
lb,(bb-ub)*k/5+ub,lb+5,(bb-ub)*k/5+ub);
}

/*draw the outline of the graph*/
XSetLineAttributes(mydisplay,mygc,3,LineSolid,
CapRound,JoinRound);
XDrawLine(myevent.xexpose.display,
myevent.xexpose.window,  mygc,lb,bb,rb,bb);
XDrawLine(myevent.xexpose.display,
myevent.xexpose.window,  mygc,lb,ub,rb,ub);
XDrawLine(myevent.xexpose.display,
myevent.xexpose.window,  mygc,lb,ub,lb,bb);
XDrawLine(myevent.xexpose.display,
myevent.xexpose.window,  mygc,rb,ub,rb,bb);

XDrawImageString(myevent.xexpose.display,
myevent.xexpose.window,  mygc,
(1000-XTextWidth(figurefontstruct,sbt,
strlen(sbt)))/2,650,sbt,  strlen  (sbt)-1  );

XDrawImageString(myevent.xexpose.display,
myevent.xexpose.window,  mygc,
(125-XTextWidth(figurefontstruct,slt,
strlen(slt)))/2,340,slt,  strlen  (slt)-1  );

/*print  titles*/
XSetFont  (mydisplay,mygc,titlefontstruct->fid);
XDrawImageString(myevent.xexpose.display,
```

```
myevent.xexpose.window,  mygc,
(1000-XTextWidth(titlefontstruct,ut,
strlen(ut)))/2,  40,ut,  strlen  (ut)-1  );
XDrawImageString(myevent.xexpose.display,
myevent.xexpose.window,  mygc,
(1000-XTextWidth(titlefontstruct,bt,
strlen(bt)))/2,690,bt,  strlen  (bt)-1  );

/*  process  keyboard  mapping  changes  */
case  MappingNotify:
XRefreshKeyboardMapping  (  &myevent  );
break;

/*  process  keyboard  input  */
case  KeyPress:
i  =  XLookupString  (  &myevent,  text,  10,  &mykey,
0  );
if  (  i  ==  1  &&  text[0]  ==  'q'  )  done  =  1;
break;
}    /*  switch  (myevent.type)  */
}  /*  while  (done  ==  0  )  */

/*    termination    */
XFreeGC  (mydisplay,  mygc  );
XDestroyWindow  (  mydisplay,  mywindow  );
XCloseDisplay  (  mydisplay  );
exit  (0);
}


void    nrerror(error_text)
char    error_text[];
{
fprintf(stderr,"Numerical  Recipes  run-time  error...\n");
fprintf(stderr,"%s\n",error_text);
fprintf(stderr,"...now  exiting  to  system...\n");
exit(1);
}

int    *ivector(nl,nh)
int    nl,nh;
{
int  *v;

v=(int    *)malloc((unsigned)(nh-nl+1)*sizeof(int));
if(!v)  nrerror("allocation  failure  in  ivector()");
return    v-nl;
}

float    *vector(nl,nh)
int    nl,nh;
{
float  *v;
```

```
v=(float *)malloc((unsigned)(nh-nl+1)*sizeof(float));
if(!v) nrerror("allocation failure in vector()");
return v-nl;
}


float **matrix(nrl,nrh,ncl,nch)
int nrl,nrh,ncl,nch;
{
int i;
float **m;

m=(float **)malloc((unsigned)(nrh-nrl+1)*sizeof(float*));
if(!m) nrerror("allocation failure 1 in matrix()");
m -= nrl;
for(i=nrl;i<=nrh;i++) {
m[i]=(float *)malloc((unsigned)(nch-ncl+1)*sizeof(float));
if(!m[i]) nrerror("allocation failure 2 in matrix()");
m[i] -= ncl;
}
return m;
}
```