

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

DESENVOLVIMENTO DE UMA NOVA FERRAMENTA CAD
PARA O ESTUDO DE COMPATIBILIDADE ELETROMAGNÉTICA
USANDO TRANSMISSION-LINE MATRIX - JOHN'S SUPER
NODE (TLM-JSN)

CARLOS ALBERTO TENÓRIO DE CARVALHO JÚNIOR

ORIENTADOR:
LEONARDO RODRIGUES ARAÚJO XAVIER DE MENEZES

TESE DE DOUTORADO EM ENGENHARIA ELÉTRICA

PUBLICAÇÃO: PPGENE.TD - 008A/06
BRASÍLIA/DF: MARÇO/2006

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

**DESENVOLVIMENTO DE UMA NOVA FERRAMENTA CAD
PARA O ESTUDO DE COMPATIBILIDADE ELETROMAGNÉTICA
USANDO TRANSMISSION-LINE MATRIX - JOHN'S SUPER
NODE (TLM-JSN)**

CARLOS ALBERTO TENÓRIO DE CARVALHO JÚNIOR

**TESE DE DOUTORADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA
ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE
BRASÍLIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO
DO GRAU DE DOUTOR.**

APROVADA POR:

Leonardo Rodrigues Xavier De Menezes, PhD, ENE/UnB
(Orientador)

Adroaldo Raizer, Dr., EEL/UFSC
(Examinador Externo)

Horácio Tertuliano Santos Filho, PhD, DELT/UFPR
(Examinador Externo)

Antonio José Martins Soares, Doutor, ENE/UnB
(Examinador Interno)

Paulo Henrique Portela de Carvalho, Docteur, ENE/UnB
(Examinador Interno)

BRASÍLIA/DF, 31 DE MARÇO 2006.

FICHA CATALOGRÁFICA

DE CARVALHO JR., CARLOS ALBERTO TENÓRIO

Desenvolvimento de uma Nova Ferramenta CAD para o Estudo de Compatibilidade Eletromagnética Usando Transmission-Line Matrix - John's Super Node (TLM-JSN) [Distrito Federal] 2006.

xiii, 107p., 297 mm (ENE/FT/UnB, Doutor, Engenharia Elétrica, 2006).

Tese de Doutorado - Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Elétrica.

- | | |
|---------------|-------------------------|
| 1. TLM | 2. TLM-JSN |
| 3. CAD | 4. Circuitos Integrados |
| I. ENE/FT/UnB | II. Título (série) |

REFERÊNCIA BIBLIOGRÁFICA

DE CARVALHO JR., C. A. T. (2006). Desenvolvimento de uma Nova Ferramenta CAD para o Estudo de Compatibilidade Eletromagnética Usando Transmission-Line Matrix - John's Super Node (TLM-JSN). Tese de Doutorado, Publicação PPGENE.TD-008A/06, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 107p.

CESSÃO DE DIREITOS

NOME DO AUTOR: Carlos Alberto Tenório de Carvalho Júnior

TÍTULO: Desenvolvimento de uma Nova Ferramenta CAD para o Estudo de Compatibilidade Eletromagnética Usando Transmission-Line Matrix - John's Super Node (TLM-JSN)

GRAU: Doutor

ANO: 2006

É concedida à Universidade de Brasília permissão para reproduzir cópias desta tese de doutorado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta tese de doutorado pode ser reproduzida sem a autorização por escrito do autor.

Carlos Alberto Tenório de Carvalho Júnior
Rua Herbert de Azevedo, 2340, São Cristóvão
78902-210 - Porto Velho - RO

DEDICATÓRIA

*À minha esposa Dalva pela paciência
e aos meus pais, Carlos e Lôide Carvalho,
pelo inestimável apoio.*

AGRADECIMENTOS

Ao meu orientador Prof. Leonardo R. A. X. de Menezes, pela motivação, confiança e sua enorme contribuição nas discussões sobre a pesquisa.

Aos colaboradores da UFRGS, professores Renato Perez Ribas e André Inácio Reis, pela orientação dos trabalhos, e ao aluno de doutorado Felipe S. Marques, pela ajuda na implementação do código C++/OPENGL.

Agradeço ao CNPq pelo suporte financeiro durante toda a pesquisa.

Agradeço a família Chaiben, pela ajuda, palavras de incentivo e a inestimável amizade.

Ao amigo Leandro Santana por suas sugestões, e Nathalie C. Pinheiro pela sua contribuição nas discussões e implementações no código Matlab.

Finalmente e em especial, agradeço a minha esposa Dalva M. Seewald de Carvalho pelo carinho e paciência, e aos meus pais, Carlos e Lôide Carvalho, que sempre estiveram presentes em todos os momentos.

E todos aqueles que direta ou indiretamente contribuíram para a concretização desta pesquisa.

Muito Obrigado.

DESENVOLVIMENTO DE UMA NOVA FERRAMENTA CAD PARA O ESTUDO DE COMPATIBILIDADE ELETROMAGNÉTICA USANDO TRANSMISSION LINE MATRIX - JOHN'S SUPER NODE (TLM-JSN)

Autor: Carlos Alberto Tenório de Carvalho Júnior

Orientador: Leonardo Rodrigues Araújo Xavier De Menezes

Programa de Pós-graduação em Engenharia Elétrica

Brasília, Março de 2006.

RESUMO

Este trabalho apresenta um novo conceito para a aceleração do método TLM bidimensional (TLM-2D). A técnica baseia-se na utilização de saltos controlados no passo de tempo em conjunto com a redução de ordem de modelo, obtendo assim um processamento mais rápido. Estes saltos são numericamente estáveis e podem resultar em considerável redução na utilização do processador. Para tornar o método mais eficiente emprega-se a redução de ordem de modelo por meio da decomposição de Schur ou via decomposição em autovalores, utilizando a transformada Z. Além disso, apresenta-se aplicações da envoltória complexa na modelagem numérica utilizando o método TLM-JSN (*Transmission Line Matrix - John's Super Node*) e adaptação da técnica de diakópticas na modelagem TLM-JSN. A metodologia é validada através de comparações com o método TLM convencional, mediante o cálculo de diversas estruturas.

DEVELOPMENT OF A NEW TOOL CAD FOR THE STUDY OF ELECTROMAGNETIC COMPATIBILITY USING TRANSMISSION-LINE MATRIX - JOHN'S SUPER NODE (TLM-JSN)

Author: Carlos Alberto Tenório de Carvalho Júnior

Supervisor: Leonardo Rodrigues Araújo Xavier De Menezes

Programa de Pós-graduação em Engenharia Elétrica

Brasília, March of 2006.

ABSTRACT

This work presents a new concept for the acceleration of the TLM method. The technique is based on the use of controlled timestep jumps together with model order reduction. This results in faster processing. These jumps are numerically stable and can result in considerable computational savings. The method can be optimized through model order reduction using Schur decomposition or through Z-transform eigenvalue decomposition. In addition, this work presents the application of the complex envelope in TLM-JSN (Transmission-Line Matrix - John's Super Node) together with the application of diatoptics techniques. The methodology is validated through comparisons with the method TLM conventional, by the calculation of several structures.

Sumário

Sumário	viii
Lista de Figuras	xi
1 INTRODUÇÃO	1
1.1 PROBLEMÁTICA, ESTADO DA ARTE E MOTIVAÇÃO	1
1.2 ORGANIZAÇÃO DA TESE	3
2 FUNDAMENTOS TEÓRICOS DO MÉTODO TLM	5
2.1 INTRODUÇÃO	5
2.2 DESCRIÇÃO HISTÓRICA	5
2.3 PRINCÍPIO DE HUYGENS EM SUA FORMA DISCRETA	6
3 MÉTODO TRANSMISSION LINE MODELING	9
3.1 INTRODUÇÃO	9
3.2 LINHAS DE TRANSMISSÃO EM REGIME SENOIDAL	9
3.3 MÉTODO TLM UNIDIMENSIONAL E BIDIMENSIONAL	11
3.3.1 TLM Unidimensional - 1D	11
3.3.2 TLM Bidimensional - 2D	14
3.3.3 Modelagem de perdas em materiais não-homogêneos	21
4 MÉTODO TRANSMISSION LINE MODELING – JOHN’S SUPER NODE	25
4.1 INTRODUÇÃO	25
4.2 MODELAGEM 2D DO MÉTODO TLM-JSN	25
4.3 MODELAGEM MATRICIAL	26
4.3.1 Matriz de Espalhamento	27

4.3.2	Matriz de Conectividade	32
4.3.3	Matriz de Transmissão	33
4.4	INTERPOLAÇÃO LINEAR DAS EQUAÇÕES DE ESTADO DO TLM-JSN	34
4.5	REDUÇÃO DE ORDEM DO MODELO	36
4.5.1	Primeira Redução de Modelo - ROM 1	37
4.5.2	Segunda Redução de Modelo - ROM 2	38
4.6	FLUXOGRAMA DO MÉTODO TLM-JSN	39
5	APLICAÇÃO DA ENVOLTÓRIA COMPLEXA NO MÉTODO TLM-JSN	41
5.1	INTRODUÇÃO	41
5.2	APLICAÇÃO DA ENVOLTÓRIA COMPLEXA EXATA NO MÉTODO TLM-JSN	41
5.3	APLICAÇÃO DA ENVOLTÓRIA COMPACTA NO TLM-JSN	45
6	MODELAGEM TLM-JSN UTILIZANDO TÉCNICAS DE SEGMENTAÇÃO	49
6.1	INTRODUÇÃO	49
6.2	DIAKOPTICAS NA MODELAGEM TLM-JSN	49
7	SIMULAÇÕES E RESULTADOS	54
7.1	INTRODUÇÃO	54
7.2	ESTRUTURAS ANALISADAS	55
7.2.1	Filtro WR28	55
7.2.2	Filtro IRIS Tipo A	59
7.2.3	Filtro IRIS Tipo B	62
7.2.4	Envoltória Complexa	64
7.2.5	Diakopticas	66
8	CONCLUSÃO	69
8.1	PROPOSTA DE DESENVOLVIMENTO FUTURO	70
	REFERÊNCIAS BIBLIOGRÁFICAS	71
	APÊNDICES	76

A FERRAMENTA E CÓDIGO USADO NA PESQUISA	77
A.1 PRODUTO TENSORIAL DE KRONNECKER NA OBTENÇÃO DA MATRIZ DE ESPALHAMENTO	77
A.1.1 Definição	77
A.1.2 Decompondo a Matriz S em Produto Tensorial de Kronecker . . .	77
A.1.3 Proposição	78
A.2 CÓDIGO EM MATLAB DO TLM-JSN BIDIMENSIONAL	79
A.2.1 IRIS 1	79
A.2.2 IRIS 2	79
A.2.3 IRIS 3	80
A.2.4 JSN PURO	80
A.2.5 JSN ROM 1	84
A.2.6 JSN ROM 2	89
A.3 ARTIGOS PUBLICADOS	93
A.3.1 Periódicos Indexados	93
A.3.2 Conferência Internacional	93
A.3.3 Conferência Nacional	93

Lista de Figuras

1.1	Diagrama organizacional deste trabalho.	3
2.1	Princípio de Huygens e formação das frentes de ondas secundárias [9]. . . .	7
2.2	(a) Representação simbólica de um espaço bidimensional discretizado (a e b), e modelo equivalente usando linhas de transmissão (c e d). (a) Campo incidente em um ponto no espaço. (b) Espalhamento do impulso. (c) Impulso de tensão incidente em um nó equivalente em uma malha cartesiana de linhas de transmissão. (d) Espalhamento da tensão incidente em um nó.	8
3.1	Linha de Transmissão	10
3.2	Função de impedância $Z(x)$ de uma linha de transmissão variante.	11
3.3	Função de impedância $Z(x)$ discretizada em N elementos	12
3.4	Representação de uma linha de transmissão discretizada em N elementos .	12
3.5	Fenômeno de propagação na Interface n	12
3.6	Fluxograma do algoritmo TLM-1D.	14
3.7	Estrutura discretizada por um nó TLM série	15
3.8	(a) Estrutura de um nó TLM série com quatro linhas de transmissão e; (b) sua simplificação	16
3.9	Análise da propagação em um nó TLM: (a) pulso de tensão incidente e (b) primeiro evento de espalhamento.	16
3.10	Circuito equivalente de Thevenin para o nó serie	17
3.11	Conexão em um nó TLM série	19
3.12	Fluxograma do algoritmo TLM-2D	20
3.13	Nó paralelo no método TLM bidimensional	21
3.14	Modelo do nó paralelo com <i>stub</i>	22
3.15	Circuito equivalente por Thevenin.	23

4.1	Super nó formado pela união de quatro nós adjacentes.	27
4.2	Super nó formado por quatro nós adjacentes	27
4.3	Obtenção das Matrizes $[C]$ e $[T]$ para o super nó de quatro nós.	32
4.4	Fluxograma do algoritmo TLM-JSN.	40
6.1	Representação em blocos de técnica de segmentação no método TLM bidimensional.	50
6.2	Representação das perdas de energia na aplicação da técnica de diakoptica no método TLM-JSN	52
6.3	Fluxograma do algoritmo TLM-JSN.	53
7.1	Amplitude do pulso gaussiano e sua Magnitude	54
7.2	Filtro utilizado para a simulação TLM	55
7.3	Comparação entre as TLM Normal, TLM (MEFISTO) e TLM-JSN ($p = 2$), para a estrutura da Fig. 7.2	56
7.4	Comparação do TLM-JSN para diversas razões p para a estrutura da Fig. 7.2	56
7.5	Comparação do TLM-JSN com ROM-2 e o TLM Normal para a estrutura da Fig. 7.2	57
7.6	Comparação dos tempos de processamento entre as versões do TLM-JSN e do TLM normal para a estrutura da Fig. 7.2	58
7.7	Espectro de Frequências da resposta obtida para a estrutura da Fig. 7.2, ($p = 2$)	58
7.8	Filtro Íris utilizado para a simulação TLM.	59
7.9	Comparação entre o TLM Normal, TLM -JSN ($p = 16$) e TLM-JSN-ROM-2 ($p = 16$), para a estrutura da Fig. 7.8	60
7.10	Comparação do TLM-JSN sem ROM para diversas razões p , para a estrutura da Fig. 7.8	60
7.11	Comparação dos tempos de processamento entre as versões do TLM-JSN e do TLM normal, para a estrutura da Fig.7.8	61
7.12	Espectro de Frequências da resposta obtida para a estrutura da Fig. 7.8, ($p = 30$)	61
7.13	Tipo de Filtro Íris utilizado para a simulação TLM.	62

7.14	Comparação do TLM-JSN-ROM-1 para diversas razões p , para a estrutura da Fig.7.13	63
7.15	Comparação dos tempos de processamento entre as versões do TLM-JSN com ROM e do TLM normal, para a estrutura da Fig. 7.13	63
7.16	Espectro de Frequências da resposta obtida para a estrutura da Fig. 7.13, ($p = 10$)	64
7.17	Resposta em amplitude para o guia de onda WR28 com a técnica de envelope complexo. As curvas I e II foram obtidas com envelope completo e as curvas III e IV com o compacto.	65
7.18	Resposta em amplitude para o espaço vazio com a técnica de envelope complexo. As curvas I e II foram obtidas com envelope completo e as curvas II e IV com o compacto.	65
7.19	Estruturas simuladas com TLM e TLM-JSN com perdas.	66
7.20	Resposta em Amplitude para a estrutura da Fig. 7.19, usando TLM e TLM-JSN com perdas com diferentes razões de salto.	67
7.21	Resposta em Amplitude usando TLM e TLM-JSN com perdas e técnica de diakópticas. Os blocos possuem 5×5 células cada.	67
7.22	Estrutura simulada com TLM e TLM-JSN com perdas e técnica de diakópticas usando quatro blocos, possuindo 5×5 células cada com $\Delta x = \Delta y = 0,05\mu m$	68
7.23	Resposta em Amplitude usando TLM e TLM-JSN com perdas e técnica de diakópticas usando quatro blocos.	68

1– INTRODUÇÃO

1.1– PROBLEMÁTICA, ESTADO DA ARTE E MOTIVAÇÃO

A análise de estruturas eletromagnéticas é um ramo da engenharia elétrica muito pesquisado atualmente. Diversas técnicas são utilizadas, cada uma com suas características e tempos de respostas específicos. Dependendo da técnica utilizada, a simulação completa de uma estrutura mais complexa pode se tornar demasiadamente demorada e às vezes a solução obtida não será a desejada. Um método numérico que vem sendo muito utilizado nos últimos anos é o TLM (*Transmission Line Modeling*) para simulações de vários tipos de problemas eletromagnéticos [1].

O método TLM é uma técnica numérica que permite a resolução das equações de Maxwell para o caso mais geral de propagação de ondas eletromagnéticas no domínio do tempo [2–6]. Criado no início dos anos 70 por P. B. Johns, o TLM tem sido utilizado intensamente nas duas últimas décadas graças às suas características peculiares aplicáveis a um grande número de problemas. Há várias aplicações do TLM em eletromagnetismo. Entre elas destacam-se: estudo da compatibilidade eletromagnética (EMC) e interferência eletromagnética (EMI) de dispositivos; modelagem de antenas; projeto de radares; determinação do aquecimento eletromagnético; simulação da propagação dos campos eletromagnéticos no corpo humano, entre outras. Em outras áreas do conhecimento o método TLM também é utilizado, tem-se como exemplo o estudo de sistemas mecânicos, acústicos e de difusão térmica. Diferentemente de outros métodos numéricos no domínio do tempo, onde as equações de Maxwell são resolvidas diretamente no meio em estudo, o TLM baseia-se no uso de redes de circuitos elétricos para a solução de problemas de espalhamento, segundo a teoria ondulatória da luz ou princípio de Huygens.

Este método pode ser empregado para a resolução de redes simples, bem como estrutu-

ras de modelagem mais complexa. Entretanto, como outros métodos, o TLM também apresenta algumas desvantagens. Por exemplo, no caso particular da análise eletromagnética de circuitos de microeletrônica devido à relação entre a discretização espacial e a temporal, a análise de um circuito integrado irá exigir um refinamento na discretização e conseqüentemente um aumento demasiado no tempo de simulação.

Este problema relacionado com a discretização espacial e temporal desencadeou a proposta desta pesquisa, pois está vinculada ao Instituto do Milênio em parceria com a Embrapa no desenvolvimento de um sistema de comunicação em *chip*. Este *chip* se insere no projeto de um sistema de baixo custo para automação do processo de irrigação de pequenas propriedades. O *chip* recebe os dados provenientes de um sensor de umidade enterrado no solo e os envia através de sinais de RF para a Estação de Campo, à qual avalia a necessidade de acionamento do sistema de irrigação. Os dados de comando do sistema de irrigação são enviados pela Estação de Campo via RF ao *chip*, que é responsável pelo acionamento do sistema de irrigação [7]. Com isso, a busca por uma solução mais adequada para o problema de simulações em circuitos integrados vem sendo pesquisada.

O principal foco desta pesquisa é no desenvolvimento de uma ferramenta capaz de fazer análise eletromagnética de circuitos integrados para o estudo de compatibilidade e interferência eletromagnética. Porém, este objetivo tornou-se extenso à medida que a pesquisa avançou. Contudo foi atingido um ponto satisfatório no desenvolvimento de uma nova modelagem para o método numérico escolhido, no caso o TLM, para obter um menor tempo de processamento. Nesta pesquisa obteve-se uma versão acelerada do TLM (*Transmission Line Modeling*) bidimensional (TLM-2D).

A metodologia utilizada neste trabalho fica mais clara quando observada a Fig. 1.1. Primeiramente foi adotado um método, após sua escolha foi feita uma contribuição original à este método, em que adota-se a modelagem em super blocos e o uso de técnica de aceleração. Além disso, foi implementada técnica de segmentação ou diakóptica para divisão espacial do problema.

Apresenta-se de forma objetiva nos capítulos seguintes toda a teoria envolvida com a formulação e os passos matemáticos para o desenvolvimento deste novo conceito em aceleração no domínio do tempo.

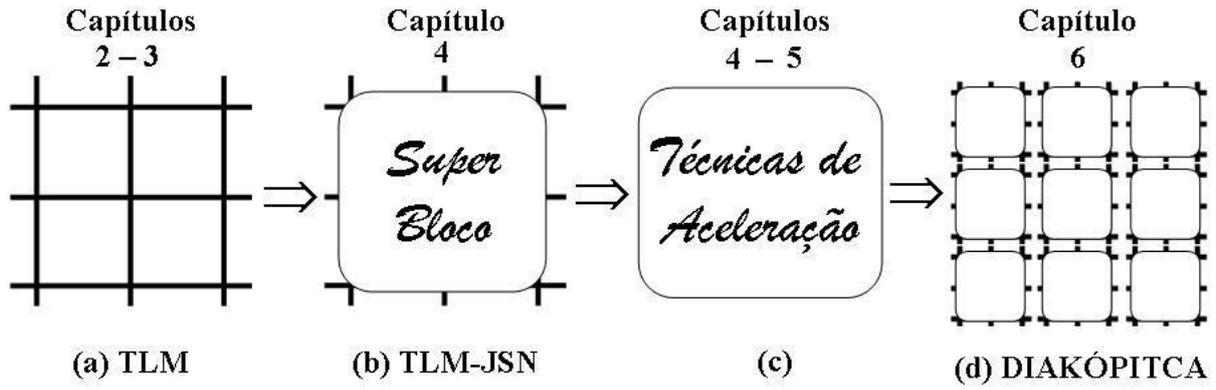


Figura 1.1: Diagrama organizacional deste trabalho.

Na próxima seção é descrita a organização deste trabalho e um breve resumo sobre o que trata cada capítulo.

1.2– ORGANIZAÇÃO DA TESE

Esta tese foi elaborada em nove capítulos com a intenção de esclarecer de forma resumida e objetiva sobre a teoria envolvida nesta pesquisa, bem como auxiliar na melhor compreensão do método TLM em seus aspectos mais básicos.

Um breve histórico do desenvolvimento do método TLM é apresentado no Capítulo 2 bem como a fundamentação teórica que deu origem à sua formulação.

O Capítulo 3 mostra de forma resumida a teoria básica do método TLM. Além disso, apresenta de forma sucinta a teoria de Linhas de Transmissão, passando pelo TLM unidimensional e finalizando com o TLM bidimensional.

Nos Capítulos 4, 5 e 6, concentram-se todos os esforços realizados para obtenção de um resultado original. O novo algoritmo do TLM desenvolvido neste trabalho é apresentado no Capítulo 4. Toda sua concepção, incluindo o procedimento para obtenção das matrizes, a descrição do algoritmo de passos de tempo controlados e as reduções de ordem de modelo são descritos.

Afim de consolidar a teoria para atingir os objetivos mencionados na seção anterior apresenta-se o Capítulo 5, que trata sobre a aplicação da envoltória complexa no método TLM-JSN, e o Capítulo 6, que descreve a utilização de segmentação no método

TLM-JSN.

O Capítulo 7, por sua vez, mostra as simulações realizadas com o novo algoritmo apresentado e complementado com os resultados das técnicas desenvolvidas e implementadas. As análises eletromagnéticas foram realizadas em diversas estruturas para verificação e validação do método. Estão presentes também comparações entre as versões da técnica criada.

A conclusão encontra-se no Capítulo 8, bem como propostas de implementações e desenvolvimentos futuros. Ainda há um apêndice contendo a formulação do produto de Kronecker, referências dos artigos publicados durante o desenvolvimento desta pesquisa e o pseudo-código básico em Matlab.

2– FUNDAMENTOS TEÓRICOS DO MÉTODO TLM

2.1– INTRODUÇÃO

A técnica TLM foi desenvolvida por P.B. Johns e difundida em meados de 1970, e ficou conhecida como *Transmission Line Matrix Method* ou *Transmission Line Modeling Method* [8]. A fundamentação teórica é descrita neste capítulo, bem como o algoritmo básico para simulação de propagação de ondas em espaços bidimensionais com e sem fronteiras. O método TLM é apropriado para resolver problemas de espalhamento de ondas em vários aspectos; pode envolver não-linearidade, meios não-homogêneos, anisotrópicos, materiais com propriedades dependentes do tempo e diversos tipos de geometria.

2.2– DESCRIÇÃO HISTÓRICA

Em meado do século XVII, dois modelos distintos que descrevem o fenômeno da luz foram desenvolvidos: o modelo corpuscular de Isaac Newton e o modelo de ondas de Christiaan Huygens [9]. Na época de suas concepções, esses modelos foram considerados incompatíveis. Entretanto, a teoria quântica tem demonstrado que a luz em particular, como radiação eletromagnética, possui propriedades granular e de onda. Estes aspectos são complementares, e, um ou outro, são dominantes dependendo do fenômeno em estudo.

Nas frequências de microondas e ondas milimétricas, a natureza granular de radiação eletromagnética não é óbvia, manifestando-se somente em certas interações com a matéria, enquanto em todas as situações envolvendo propagação e espalhamento, o aspecto predominantemente é o de onda. Isto sugere que o modelo proposto por Huygens, e mais tarde refinado por Fresnel, podia dar base a métodos gerais de tratamento para problemas de propagação e espalhamento eletromagnéticos, particularmente no sucesso considerável em

descrever problemas de difração e interferência de ondas de luz.

Em 1971, Johns e Beurlle [8] descreveram uma nova técnica numérica para resolver problemas de espalhamento bidimensional, a qual foi baseada no modelo de propagação de ondas de Huygens. Ainda, inspirado na técnica de simulação por redes [10–12], este método empregou uma malha cartesiana de linhas de transmissão para simular a propagação bidimensional de impulsos de tensão. Logo em seguida, Johns e Akhtarzad [13–17] estenderam o método para meios não-homogêneos e incluíram o efeito de perda. Utilizando como trabalho base desses autores, outros pesquisadores [18–24] adicionaram vários aspectos e implementações como tamanho de malha variável, e técnicas de correção do erro.

Um outro importante desenvolvimento tem sido a introdução da técnica de diakópticas no domínio do tempo. Primeiramente introduzido na teoria de redes aplicado em equações de estado por Kron [25], a técnica de diakópticas consiste na divisão de uma larga rede em sub-redes para uma análise individual e subseqüentemente conectá-las [26, 27]. Esta técnica foi usada por Brewitt-Taylor e Johns [28] em modelagem de campos por redes e equações de estado, e foi então estendido para modelos discretos no tempo por Johns e Akhtarzad [29, 30]. Hoefler tem desenvolvido essa aproximação generalizada com a técnica da função discreta de Green, chamada de *Johns Matrix Techniques* [31–36].

Apresenta-se a seguir o estudo da versão do modelo de onda de Huygens discretizado em duas dimensões e no tempo, e foi a inspiração para a fundamentação do método TLM. Discussões mais detalhadas podem ser encontradas em um artigo muito interessante de P.B. Johns [16].

2.3– PRINCÍPIO DE HUYGENS EM SUA FORMA DISCRETA

De acordo com Huygens [9], uma frente de onda consiste de um número de radiações secundárias, as quais causam ondas esféricas. O envelope dessas ondas formam uma nova frente de onda a qual, por sua vez, gera novas ondas esféricas e assim por diante. Uma aplicação sistemática desse princípio leva a uma propagação precisa da onda e seu espalhamento, mostrado na Fig. 2.1. Nota-se que com a próxima frente de onda sua energia diminui até extinguir-se.

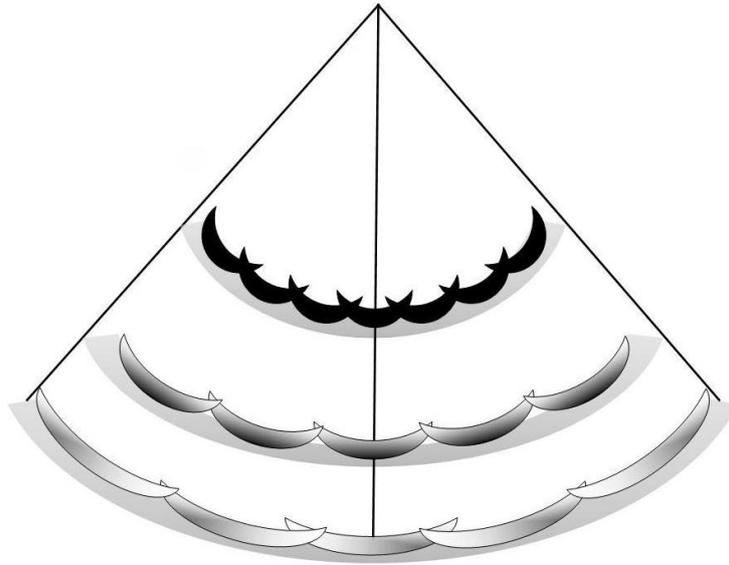


Figura 2.1: *Princípio de Huygens e formação das frentes de ondas secundárias [9].*

A fim de implementar o modelo de onda, com o princípio de Huygens em computadores, deve-se usar uma formulação na forma discretizada. Para isso, o espaço e o tempo são representados em termos finitos, Δl e Δt , os quais estão relacionados com a velocidade v das ondas elementares dadas por

$$\Delta t = \frac{\Delta l}{v} \quad (2.1)$$

O espaço bidimensional é modelado por uma malha cartesiana, separadas por Δl . A unidade do tempo Δt é o tempo necessário para uma onda viajar de um nó para um mais próximo, percorrendo Δl unidades em cada iteração.

Afim de exemplificar o princípio de Huygens, considera-se um pulso com energia eletromagnética é incidente em um nó e a amplitude de campo associada com essa energia é unitária. De acordo com o princípio de Huygens, essa energia espalha-se isotropicamente nas quatro direções, cada impulso incidente de energia percorre um dos quatro ramos. A quantidade de campo espalhado deve ser $1/2$ em magnitude (descrito em mais detalhes no Capítulo 3). Além disso, o coeficiente de reflexão visto pelo pulso incidente deve ser negativo para assegurar a continuidade do campo no nó. Esse fenômeno pode ser visto melhor na Fig. 2.2.

O método bidimensional tem uma rede análoga a uma forma de malha ortogonal de linhas de transmissão, ou seja, uma matriz de linhas, (Fig. 2.2 – c e d), formando uma matriz de

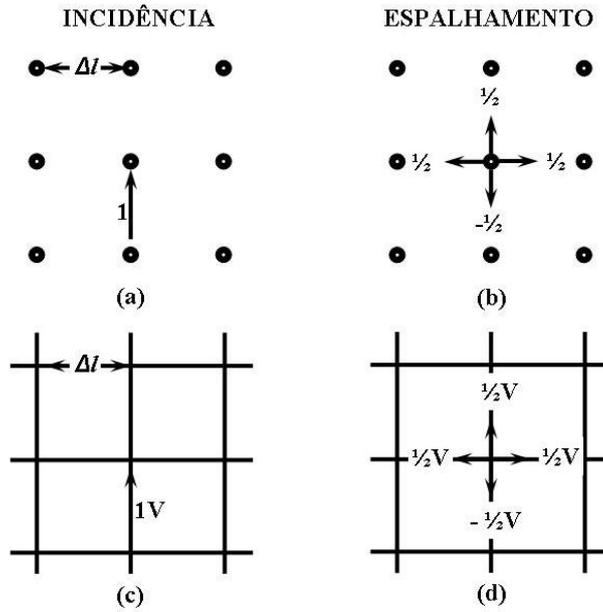


Figura 2.2: (a) Representação simbólica de um espaço bidimensional discretizado (a e b), e modelo equivalente usando linhas de transmissão (c e d). (a) Campo incidente em um ponto no espaço. (b) Espalhamento do impulso. (c) Impulso de tensão incidente em um nó equivalente em uma malha cartesiana de linhas de transmissão. (d) Espalhamento da tensão incidente em um nó.

nós que possuem as mesmas características dos nós da Fig. 2.2 – a. Além disso é possível modelar o procedimento de propagação do impulso em uma série de conexões de linhas de transmissão, pois os nós possuem essa característica dual nas conexões paralelas. Não existe o efeito de *crossstalk* entre as linhas da malha e somente acoplamento entre elas nas conexões paralelas de cada nó.

Posteriormente mostra-se que, no método TLM, as tensões e correntes são equivalentes aos campos elétrico e magnético das equações de Maxwell [8]. Isso garante a consistência do modelo TLM com as equações de Maxwell e certamente pode-se modelar bidimensionalmente a propagação de ondas.

A seguir apresenta-se de forma sucinta a teoria sobre linhas de transmissão, a qual enfatiza sobre as tensões incidentes e refletidas, e sobre o coeficiente de reflexão. Todos estes parâmetros são a base da modelagem TLM.

3– MÉTODO TRANSMISSION LINE MODELING

3.1– INTRODUÇÃO

O Método TLM baseia-se no uso de redes de circuitos elétricos para problemas de espalhamento, segundo o princípio de Huygens. É um método utilizado na resolução numérica no domínio do tempo das equações de Maxwell para casos mais abrangentes em propagação de ondas eletromagnéticas, i.e., permitindo a modelagem de vários tipos de problemas eletromagnéticos com estruturas geométricas complexas, materiais com propriedades não lineares, não homogêneas e com perdas. Além de avaliar materiais com parâmetros de dispersão (dependentes da frequência) e anisotrópicos. Neste capítulo apresenta-se a teoria sobre linhas de transmissão enfocando os aspectos mais relevantes para a modelagem TLM. Além disso, mostra-se os aspectos básicos do TLM unidimensional e bidimensional.

3.2– LINHAS DE TRANSMISSÃO EM REGIME SENOIDAL

A teoria de Linhas de Transmissão é de grande ajuda no entendimento do método TLM, principalmente quando refere-se a casamento de impedâncias. Nesta seção apresenta-se um breve resumo sobre essa teoria, que pode ser vista mais detalhada em [37].

Seja a tensão e a corrente numa linha de transmissão com uma fonte de tensão senoidal definidas para um instante t e uma distância z , como:

$$V(t, z) = |V^+| \cos\left(\omega \left[t - \frac{z}{v}\right]\right) + |V^-| \cos\left(\omega \left[t + \frac{z}{v} + \varphi_-\right]\right) \quad (3.1)$$

$$I(t, z) = \frac{1}{Z_0} \left\{ |V^+| \cos\left(\omega \left[t - \frac{z}{v}\right]\right) - |V^-| \cos\left(\omega \left[t + \frac{z}{v} + \varphi_-\right]\right) \right\} \quad (3.2)$$

Onde $|V^+|$ e $|V^-|$ são os módulos das ondas incidente e refletida na descontinuidade,

respectivamente, v é a velocidade de propagação da onda, φ_- é o ângulo entre a onda refletida e a incidente e Z_0 é a impedância característica do meio em ohms (Ω).

Na forma fasorial tem-se:

$$\bar{V}(z) = |V^+| e^{-j\beta z} + |V^-| e^{j\beta z} \quad (3.3)$$

$$\bar{I}(z) = \frac{1}{Z_0} [|V^+| e^{-j\beta z} - |V^-| e^{j\beta z}] \quad (3.4)$$

Onde $V^+ = |V^+|$ e $V^- = |V^-| e^{j\varphi_-}$ e $\beta = \frac{\omega}{v}$ é a constante de fase da linha.

Dada a Linha de Transmissão (Fig. 3.1), define-se o coeficiente de reflexão na carga Z_L pela razão da onda refletida pela onda incidente:

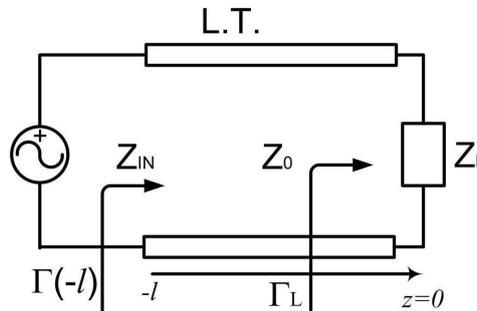


Figura 3.1: *Linha de Transmissão*

$$\Gamma_L = \frac{V^-}{V^+} = \frac{Z_L - Z_0}{Z_L + Z_0} \quad (3.5)$$

A partir desta, a impedância em uma distância $z = -l$ da carga em um meio sem perdas é:

$$\frac{\bar{V}(-l)}{\bar{I}(-l)} = Z_{IN} = Z_0 \left[\frac{e^{j\beta l} + \Gamma_L e^{-j\beta l}}{e^{j\beta l} - \Gamma_L e^{-j\beta l}} \right] = Z_0 \left[\frac{Z_L + jZ_0 \tan(\beta l)}{Z_0 + jZ_L \tan(\beta l)} \right] \quad (3.6)$$

O coeficiente de reflexão para qualquer distância z é definido por:

$$\Gamma(z) = \frac{Z_{IN} - Z_0}{Z_{IN} + Z_0} = \frac{V^- e^{j\beta z}}{V^+ e^{-j\beta z}} = \Gamma_L e^{2j\beta z} \quad (3.7)$$

Dos resultados anteriores, conclui-se que o coeficiente de reflexão é uma medida que indica se vai existir uma onda refletida ou se uma dada onda incidente chega a uma carga (descontinuidade). Se a impedância de entrada é igual à impedância da linha, observa-se que o coeficiente de reflexão é nulo, o que indica ausência de onda refletida. Fazer com que a impedância da linha seja igual à impedância da carga significa “casar a linha”, isto é, garantir que a onda incidente seja totalmente absorvida pela carga.

3.3– MÉTODO TLM UNIDIMENSIONAL E BIDIMENSIONAL

Em uma simulação TLM típica, uma malha de linhas de transmissão representa o espaço discretizado, e os campos elétrico e magnético são feitos equivalentes a tensões e correntes na malha, respectivamente. O método TLM pode ser encontrado nas versões unidimensional (1D), bidimensional (2D) e tridimensional (3D). Apresenta-se neste Capítulo uma abordagem sucinta sobre as técnicas em 1D e 2D.

3.3.1– TLM Unidimensional - 1D

A fundamentação da modelagem unidimensional foi descrita anteriormente. O principal objetivo na modelagem TLM-1D é encontrar as tensões incidentes e refletidas, bem como o coeficiente de reflexão. Para iniciar a modelagem unidimensional, considera-se uma linha de transmissão sem perdas e que possua impedância variável com comprimento. Essa variação pode ser descrita pela função de impedância $Z(x)$, Fig.3.2 [38]. Assumindo que $Z(x)$ é não nula e varia continuamente ao longo de x , de Z_0 e Z_L , valores inicial e final, respectivamente, pode-se discretizá-la em N elementos como mostrado na Fig. 3.3.

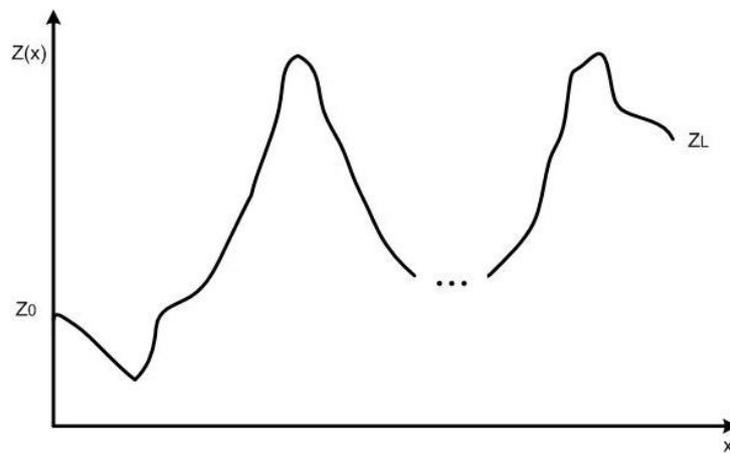


Figura 3.2: Função de impedância $Z(x)$ de uma linha de transmissão variante.

Com a função de impedância $Z(x)$ discretizada em $N\Delta x$, pode-se assumir que Δx seja muito menor que o comprimento de onda considerado ($\Delta x \ll \lambda$), de forma a minimizar os erros introduzidos pela discretização [1]. Dessa forma, para cada segmento de com-

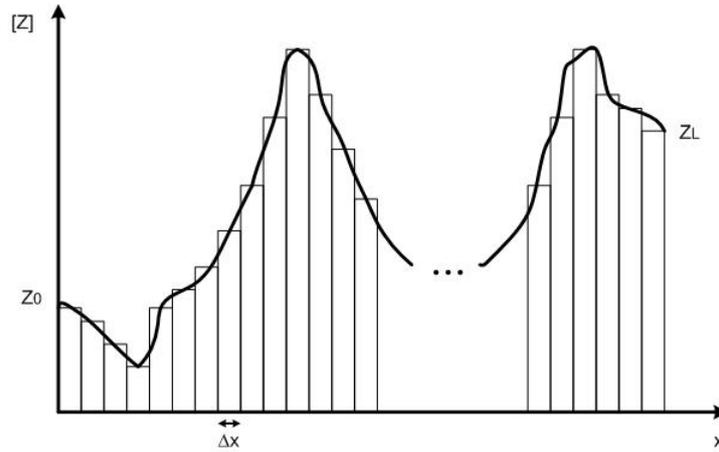


Figura 3.3: Função de impedância $Z(x)$ discretizada em N elementos

primento Δx têm um valor de impedância Z_n , onde n representa o índice do vetor de impedância $[Z]$. Esse vetor possui N elementos. A linha de transmissão discretizada é mostrada na Fig. 3.4.

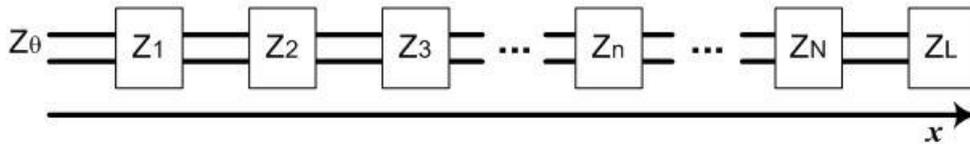


Figura 3.4: Representação de uma linha de transmissão discretizada em N elementos

Agora será analisado o fenômeno de propagação entre dois elementos, Z_{n-1} e Z_n , do perfil de impedância. Convencionando-se o sentido de propagação positivo da esquerda para a direita (ou de Z_{n-1} para Z_n), se uma onda V_{ip}^n incidir na interface entre esses dois elementos no sentido positivo e outra onda V_{in}^n incidir no sentido negativo, obter-se-ão ondas refletidas no sentido positivo V_{rp}^n e negativo V_{rn}^n como mostrado na Fig. 3.5.

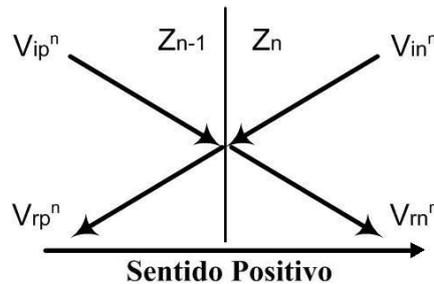


Figura 3.5: Fenômeno de propagação na Interface n

As ondas refletidas podem ser calculadas por:

$$\begin{bmatrix} V_{rp}^n \\ V_{rn}^n \end{bmatrix} = \begin{bmatrix} T_n & \Gamma'_n \\ \Gamma_n & T'_n \end{bmatrix} \begin{bmatrix} V_{ip}^n \\ V_{in}^n \end{bmatrix} \quad (3.8)$$

Onde T_n é o coeficiente de transmissão de Z_{n-1} para Z_n , Γ_n é o coeficiente de reflexão da interface entre Z_{n-1} e Z_n no sentido positivo, T'_n é o coeficiente de transmissão de Z_n para Z_{n-1} e Γ'_n é o coeficiente de reflexão da interface no sentido negativo. Estes coeficientes podem ser calculados por:

$$T_n = \frac{2Z_n}{Z_{n-1} + Z_n} \quad (3.9)$$

$$T'_n = \frac{2Z_{n-1}}{Z_{n-1} + Z_n} \quad (3.10)$$

$$\Gamma_n = \frac{Z_n - Z_{n-1}}{Z_{n-1} + Z_n} \quad (3.11)$$

$$\Gamma'_n = \frac{Z_{n-1} - Z_n}{Z_{n-1} + Z_n} \quad (3.12)$$

Com esses coeficientes pode-se empregar a técnica TLM-1D para encontrar o perfil de impedância de uma estrutura plana, por exemplo uma placa *microstrip*. O algoritmo TLM-1D é construído de forma simples. Dadas as condições iniciais V_{ip}^n e V_{in}^n em todas as interfaces, para cada passo de tempo k são calculadas as ondas refletidas V_{rp}^n e V_{rn}^n através de (3.9) a (3.12). Em seguida faz-se a conexão entre as ondas refletidas e incidentes para o próximo passo de tempo:

$${}_{k+1}V_{ip}^n = {}_k V_{rp}^{n-1} \quad (3.13)$$

$${}_{k+1}V_{in}^n = {}_k V_{rn}^{n+1} \quad (3.14)$$

No fim da linha, parte da onda que Z_N entrega a carga é refletida pelo coeficiente de reflexão Γ_L e, no próximo passo de tempo, volta a incidir sobre Z_N . Ou seja:

$${}_{k+1}V_{in}^N = \Gamma_k V_{rp}^N \quad (3.15)$$

A rotina do algoritmo pode ser representada pelo fluxograma da Fig. 3.6.

Através de técnicas de processamento de sinais, pode-se obter vários parâmetros, como: corrente, tensões e perfil de impedâncias da estrutura analisada.

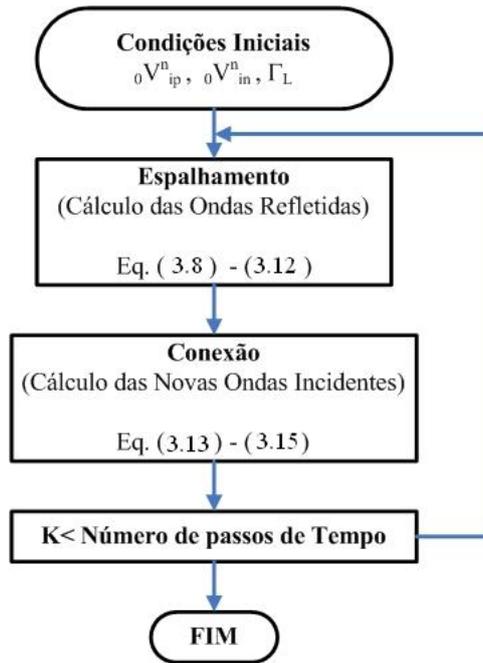


Figura 3.6: Fluxograma do algoritmo TLM-1D.

3.3.2– TLM Bidimensional - 2D

Na versão bidimensional do método TLM, emprega-se uma malha cartesiana de nós para representar o espaço discretizado. Cada nó corresponde à junção de linhas de transmissão. As ondas eletromagnéticas são representadas por impulsos de tensão que espalham-se isotropicamente sobre esta malha [1]. O objetivo deste modelo é novamente encontrar as tensões incidentes e refletidas em cada nó e, com isso, as componentes de campo elétrico e magnético. Para tal, utilizam-se as equivalências entre tensão e campo elétrico, e entre corrente e campo magnético. A configuração de campo no plano $x-y$ abordada nesta seção admite apenas componentes H_z , ou seja, devido a essa característica de propagação pode-se obter os modos de propagação transverso elétrico (TE) e transverso magnético (TM). A malha utilizada neste caso é constituída por linhas de transmissão que se interceptam em série. A Fig. 3.7 mostra uma porção do espaço representado através do nó TLM série [39].

Na Fig. 3.7, L_x, C_x, G_x e L_y, C_y, G_y são as indutâncias, capacitâncias e condutâncias para as linhas nas direções x e y respectivamente e, Δx e Δy são as dimensões do nó. Qualquer não-homogeneidade pode ser modelada por um segmento de linha de transmissão como um *stub* indutivo, L_s .

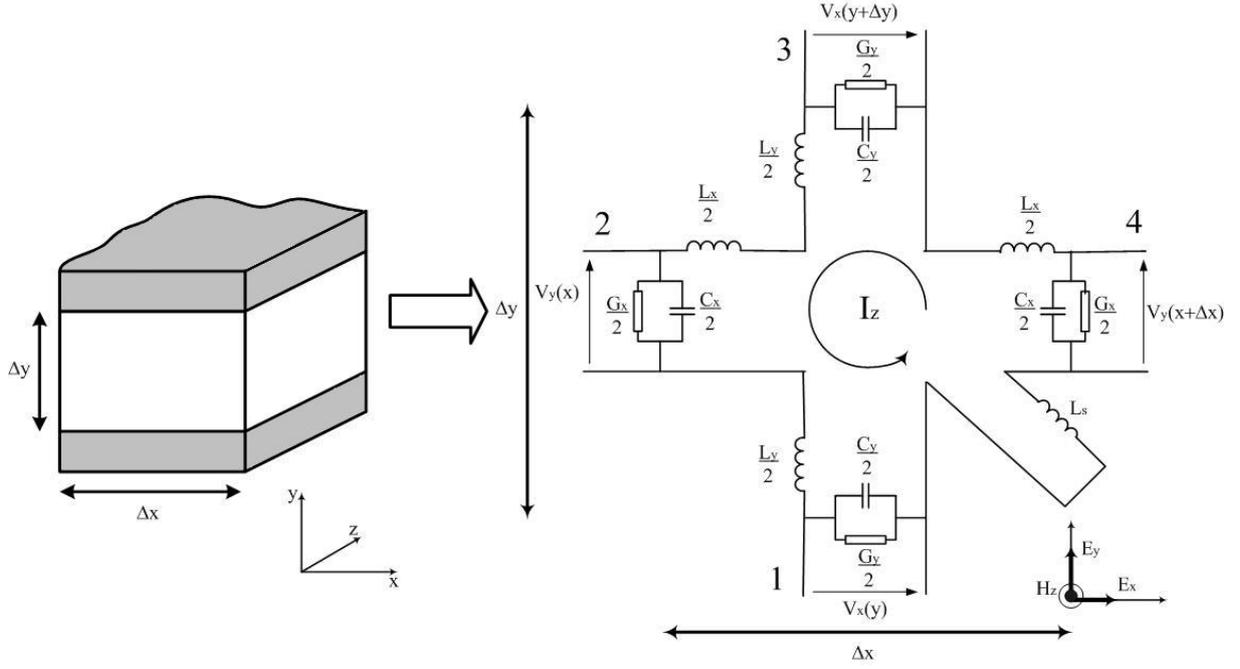


Figura 3.7: Estrutura discretizada por um nó TLM série

Para modos TE as únicas componentes de campo não nulas são: E_x , E_y e H_z . Variações de campo ao longo da direção z não são permitidas. Em um primeiro momento, desconsidera-se as perdas representadas pelas condutâncias, as equações de Maxwell resumem-se a:

$$\frac{\partial H_z}{\partial y} = \varepsilon \frac{\partial E_x}{\partial t} \quad (3.16)$$

$$-\frac{\partial H_x}{\partial x} = \varepsilon \frac{\partial E_y}{\partial t} \quad (3.17)$$

$$\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y} = \mu \frac{\partial H_x}{\partial t} \quad (3.18)$$

Onde ε e μ são a permissividade elétrica e a permeabilidade magnética do meio, respectivamente. Aplicando essas equações na estrutura discretizada da Fig. 3.7, fazendo Δx e Δy iguais a Δl e, realizando as manipulações necessárias, chega-se às seguintes equivalências entre as quantidades de campo e de circuitos:

$$\frac{I}{\Delta l} \Leftrightarrow H_z \quad (3.19)$$

$$-\frac{V_y}{\Delta l} \Leftrightarrow E_y \quad (3.20)$$

$$-\frac{V_x}{\Delta l} \Leftrightarrow E_x \quad (3.21)$$

$$L = L_y = L_x = \mu \Delta l \quad (3.22)$$

$$C = C_y = C_x = \varepsilon \Delta l \quad (3.23)$$

Neste ponto é suficiente afirmar que o nó série pode ser representado por quatro segmentos de linha de transmissão de impedância característica Z_{TL} como mostrado na Fig. 3.8.

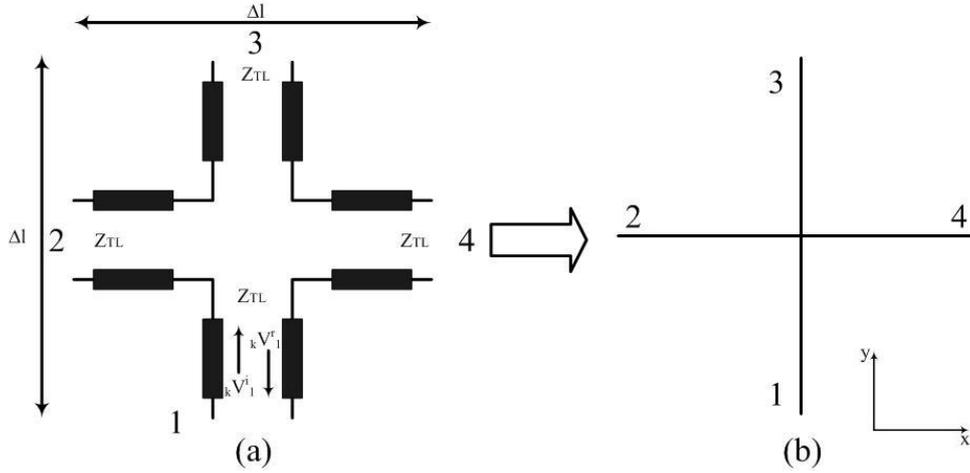


Figura 3.8: (a) Estrutura de um nó TLM série com quatro linhas de transmissão e; (b) sua simplificação

O fenômeno de propagação em uma malha TLM pode ser feito da seguinte forma. Considere um pulso de $1V$ incidente em um nó como mostrado na Fig. 3.9(a). Este pulso vai ser parcialmente transmitido e refletido de acordo com a teoria de linhas de transmissão. O pulso incidente vê três linhas de transmissão em paralelo de modo que a impedância total é $\frac{Z_{LT}}{3}$. Os sinais espalhados (transmitidos e refletidos) são mostrados na Fig. 3.9(b).

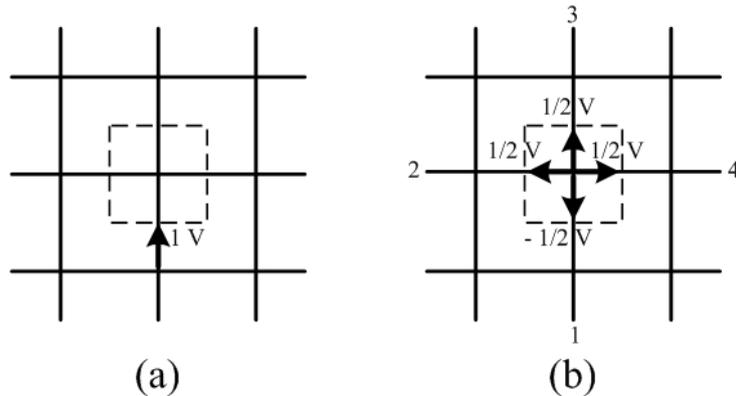


Figura 3.9: Análise da propagação em um nó TLM: (a) pulso de tensão incidente e (b) primeiro evento de espalhamento.

Os coeficientes de reflexão e transmissão são, respectivamente, iguais a:

$$\Gamma = \frac{\frac{Z_{LT}}{3} - Z_{LT}}{\frac{Z_{LT}}{3} + Z_{LT}} = -\frac{1}{2} \quad (3.24)$$

$$T = \frac{2\frac{Z_{LT}}{3}}{\frac{Z_{LT}}{3} + Z_{LT}} = \frac{1}{2} \quad (3.25)$$

A energia é conservada e espalha-se isotropicamente do nó excitado como mostrado pela equação de conservação:

$$\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2 + \left(-\frac{1}{2}\right)^2 = (1)^2 \quad (3.26)$$

Para encontrar as relações entre as tensões incidentes e refletidas, no passo de tempo k , será utilizada uma aproximação baseada na teoria de circuitos, trocando cada fração de linha de transmissão por seu circuito equivalente de Thevenin. Este consiste em uma fonte de tensão $2_k V^i$ (tensão incidente) em série com a impedância Z_{LT} como descrito em [1]. A Fig. 3.10 mostra o circuito equivalente para o nó série TLM.

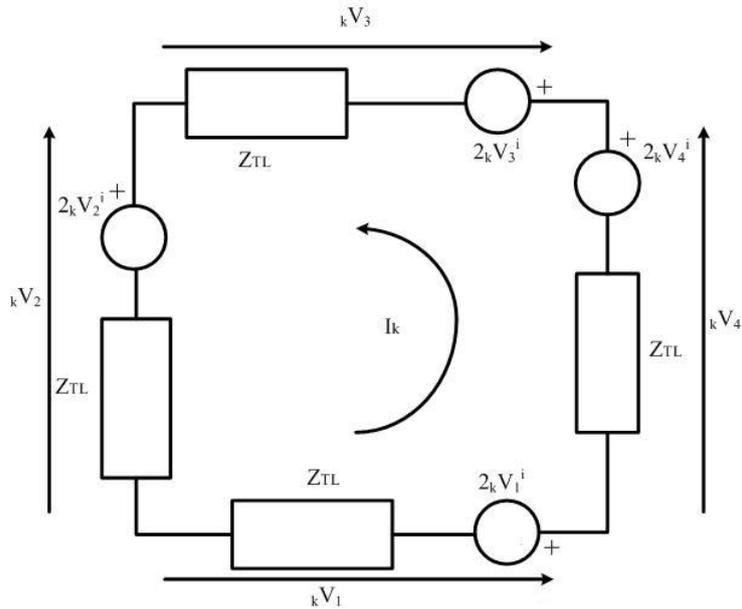


Figura 3.10: Circuito equivalente de Thevenin para o nó série

Pela Fig. 3.10 tem-se que a corrente no passo de tempo k é dada por:

$${}_k I = \frac{2_k V_1^i + 2_k V_4^i - 2_k V_2^i - 2_k V_3^i}{4Z_{LT}} \quad (3.27)$$

As tensões refletidas podem ser obtidas facilmente da expressão geral em (3.3), quando a constante de fase da linha é zero. Dessa forma:

$${}_kV^r = {}_kV - {}_kV^i \quad (3.28)$$

Onde ${}_kV$ é a tensão total no passo de tempo k . Aplicando a equação (3.28) para a tensão refletida na porta 1 (Fig. 3.8 (a)) obtém-se:

$${}_kV_1^r = {}_kV_1 - {}_kV_1^i = 2{}_kV_1^i - IZ_{LT} - {}_kV_1^i = {}_kV_1^i - I_kZ_{LT} \quad (3.29)$$

Substituindo (3.27) em (3.29), chega-se a:

$${}_kV_1^r = \frac{1}{2} ({}_kV_1^i + {}_kV_2^i + {}_kV_3^i - {}_kV_4^i) \quad (3.30)$$

A equação (3.30) pode ser estendida a outras portas de modo a se obter a matriz de espalhamento \mathbf{S} :

$${}_kV^r = \mathbf{S}_k V^i \quad (3.31)$$

Esta expressão, escrita na forma completa para o nó série, é dada abaixo:

$$\begin{bmatrix} {}_kV_1^r \\ {}_kV_2^r \\ {}_kV_3^r \\ {}_kV_4^r \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} {}_kV_1^i \\ {}_kV_2^i \\ {}_kV_3^i \\ {}_kV_4^i \end{bmatrix} \quad (3.32)$$

Para calcular as tensões incidentes no próximo passo de tempo utiliza-se a malha de nós mostrada na Fig. 3.11.

Os pulsos incidentes no nó (x, y) no passo de tempo $k + 1$ claramente são os pulsos refletidos dos nós adjacentes no passo de tempo anterior k . Ou seja, as novas tensões incidentes no nó (x, y) dependem inteiramente dos nós conectados a ele. Com isso, as seguintes expressões podem ser obtidas da Fig. 3.11:

$$\begin{aligned} {}_{k+1}V_1^i(x, y) &= {}_kV_3^r(x, y - \Delta l) \\ {}_{k+1}V_2^i(x, y) &= {}_kV_4^r(x - \Delta l, y) \\ {}_{k+1}V_3^i(x, y) &= {}_kV_1^r(x, y + \Delta l) \\ {}_{k+1}V_4^i(x, y) &= {}_kV_2^r(x + \Delta l, y) \end{aligned} \quad (3.33)$$

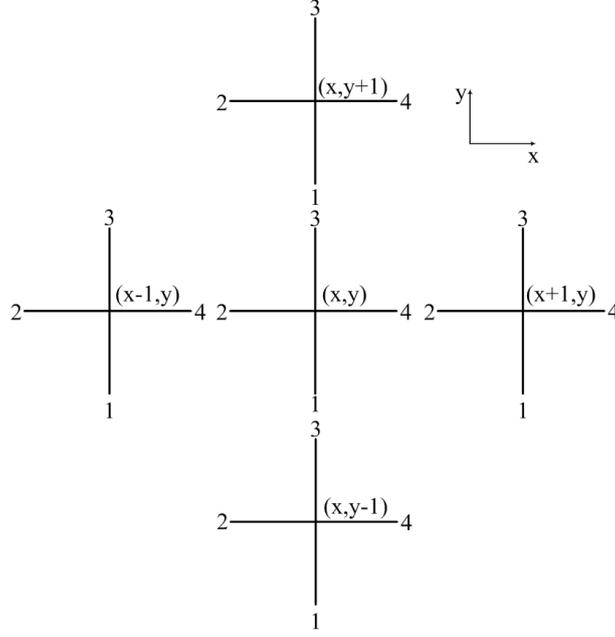


Figura 3.11: Conexão em um nó TLM série

Essas expressões representam a fase de conexão, que permitem calcular as tensões incidentes no novo passo de tempo. Para determinar as variáveis de campo através das variáveis de circuitos utiliza-se as relações expressas nas equações (3.19) a (3.23). O procedimento de cálculo inicia-se em $k = 0$ e pode ser resumido através do fluxograma apresentado na Fig. 3.12.

Nas situações práticas, a região discretizada será limitada através de fronteiras. Nesses casos as equações em (3.33) terão de ser ajustadas para incluir o efeito das condições de fronteira.

O nó paralelo, como pode ser visto na Fig. (3.13) é uma outra abordagem na modelagem TLM e também pode ser facilmente utilizado para representar a modelagem bidimensional com perdas [40]. Esse procedimento é muito similar ao apresentado na modelagem do nó série, pois ao invés de modelar no modo TE modela-se para o caso TM (componentes de campo H_x, H_y e E_z).

$$\frac{\partial E_z}{\partial y} = -\mu \frac{\partial H_x}{\partial t} \quad (3.34)$$

$$-\frac{\partial E_z}{\partial x} = -\mu \frac{\partial H_y}{\partial t} \quad (3.35)$$

$$\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} = \varepsilon \frac{\partial E_z}{\partial t} \quad (3.36)$$

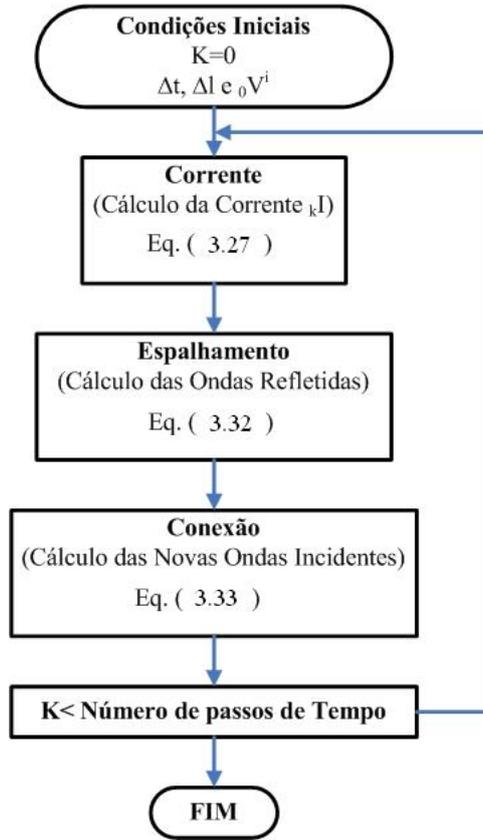


Figura 3.12: Fluxograma do algoritmo TLM-2D

Diferenciando as equações (3.34) e (3.35) com respeito a y e x , respectivamente, adicionando o resultado das equações e combinando com a equação (3.36), eliminando assim a componente de campo magnético, como mostra a equação dada por

$$\frac{\partial^2 E_z}{\partial x^2} + \frac{\partial^2 E_z}{\partial y^2} = \mu\epsilon \frac{\partial^2 E_z}{\partial t^2} \quad (3.37)$$

Essa é a equação de onda para propagação bidimensional. Como visto para o nó série a equivalência entre as componentes de campo e circuitos são verdadeiras, e descritas com mais detalhes em [1, 40]. A característica do espalhamento para o nó paralelo é a mesma para o nó série. A matriz de espalhamento para um nó paralelo é dada por

$$S = \frac{1}{2} \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix} \quad (3.38)$$

A nova tensão incidente em cada nó no próximo passo de tempo é obtida por um processo

de conexão, o qual opera exatamente como no nó série. E a lógica computacional para resolução do sistema é a mesma para o nó série.

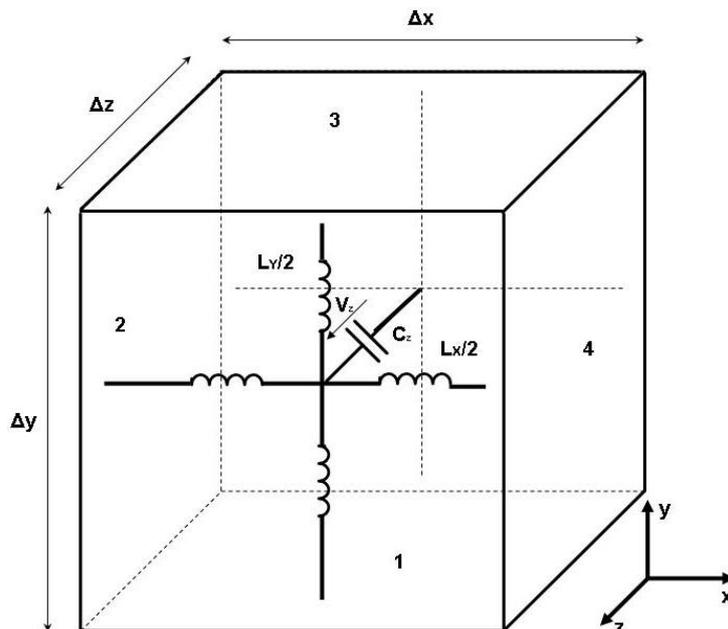


Figura 3.13: Nó paralelo no método TLM bidimensional

3.3.3– Modelagem de perdas em materiais não-homogêneos

Para incorporar a solução para materiais com perdas é necessário uma modelagem um pouco diferente da apresentada anteriormente. A princípio, a modelagem de um material não-homogêneo com ou sem perdas requer apenas modificações nos parâmetros de circuitos (C e L) para estimar o valor da permissividade dielétrica (ϵ) e permeabilidade magnética (μ). Com isso, a velocidade de propagação na malha não é mais a mesma durante todo o espalhamento e pode ser calculada por

$$u = \frac{1}{\sqrt{\mu\epsilon}} \quad (3.39)$$

No entanto, precisa-se pensar em duas possíveis condições. A primeira consiste em manter o mesmo passo de tempo (k) durante todo o problema, acarretando em um comprimento variável (Δl) na discretização dependente da velocidade de propagação u . A segunda consiste no caso inverso, onde o comprimento (Δl) na discretização é mantido, resultando em um passo de tempo (k) dependente da velocidade de propagação u . No primeiro caso,

a conectividade entre as portas da interface entre diferentes materiais não será mantida, devido ao comprimento Δl diferente para cada material. No último caso, não existe sincronismo ocasionando perda no pulso de chegada do outro lado da interface, devido Δt não ser o mesmo. Portanto, uma outra aproximação é necessária, onde a conectividade e sincronismo sejam preservados. Para conseguir isto é necessário introduzir uma capacitância e indutância adicional em forma de *stub* para obter o valor da permissividade dielétrica (ϵ) e permeabilidade magnética (μ), respectivamente.

Uma modelagem mais detalhada pode ser encontrada em um artigo muito interessante de P. B. Johns [14]. Nesta seção utiliza-se a modelagem de perdas usando o nó paralelo, a Fig. 3.14 mostra o nó com o *stub*. O procedimento é o mesmo para o nó série. Considera-se que as linhas possuem impedância característica Z_{TL} , e o *stub* capacitivo e de perdas tendo características de admitância e condutância Y_S e G_S respectivamente.

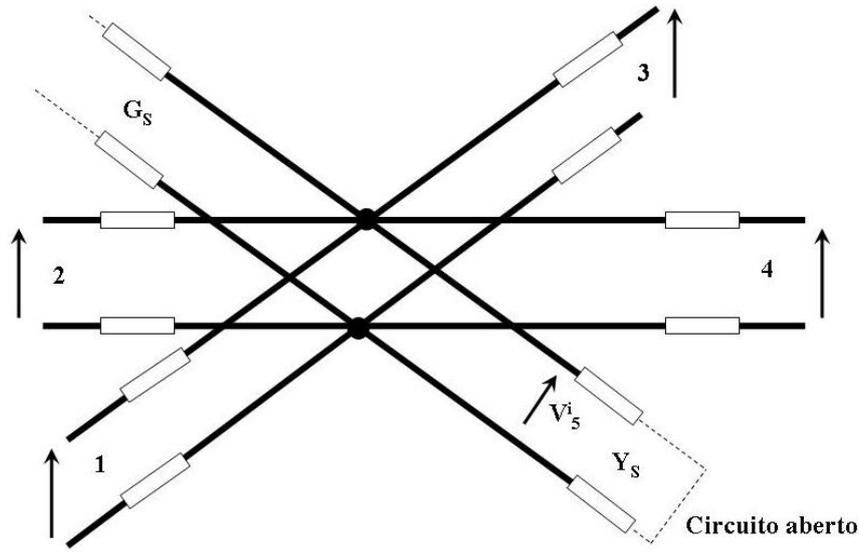


Figura 3.14: Modelo do nó paralelo com *stub*.

Normalizando a impedância e condutância usando Z_{TL} como base. Assim, cada linha tem sua impedância normalizada para 1, e a admitância dos *stubs* também por $\hat{Y} = Y_S/Z_{TL}^{-1}$ e $\hat{G} = G_S/Z_{TL}^{-1}$. O circuito equivalente por Thevenin é mostrado na Fig. 3.15, com as quantidades já normalizadas. A tensão no nó é então

$$V_z = \frac{2(V_1^i + V_2^i + V_3^i + V_4^i) + 2V_5^i \hat{Y}_S}{4 + \hat{Y}_S + \hat{G}_S} \quad (3.40)$$

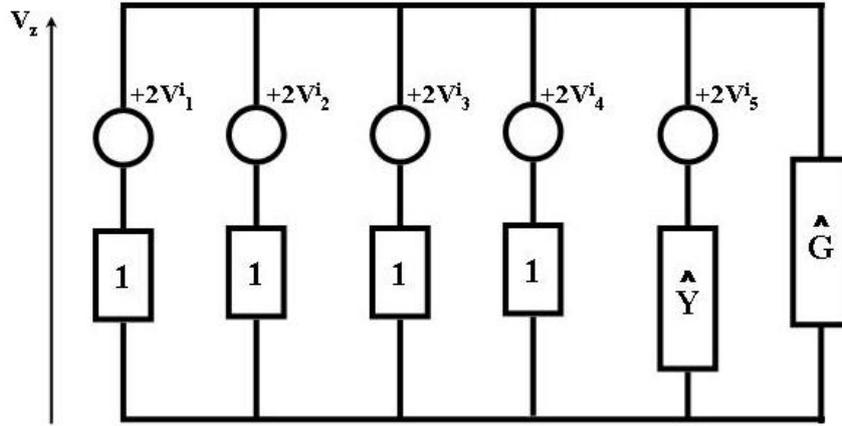


Figura 3.15: Circuito equivalente por Thevenin.

e a matriz de espalhamento é dada por

$$S = \frac{1}{\hat{Y}} \begin{bmatrix} 2 - \hat{Y} & 2 & 2 & 2 & 2\hat{Y}_S \\ 2 & 2 - \hat{Y} & 2 & 2 & 2\hat{Y}_S \\ 2 & 2 & 2 - \hat{Y} & 2 & 2\hat{Y}_S \\ 2 & 2 & 2 & 2 - \hat{Y} & 2\hat{Y}_S \\ 2 & 2 & 2 & 2 & 2\hat{Y}_S - \hat{Y} \end{bmatrix} \quad (3.41)$$

onde $\hat{Y} = 4 + \hat{Y}_S + \hat{G}_S$.

Considerando que o nó paralelo possui a característica de impedância (Z_{TL}), admitância (Y_S) e condutância (G_S). Os parâmetros modelados podem ser obtidos partindo de $C = \Delta t/Z_{TL}$ e são dados por

$$L = Z_{TL}\Delta t \quad (3.42)$$

$$L = \frac{2\Delta t}{Z_{TL}} + \frac{\Delta t}{2}Y_S \quad (3.43)$$

Da lei de correntes de Kirchhoff

$$\frac{\partial I_x}{\partial x} + \frac{\partial I_y}{\partial y} = -C_d \frac{\partial V_z}{\partial t} - \frac{V_z}{R_d} \quad (3.44)$$

onde C_d e R_d são quantidades por comprimento, isto é,

$$C_d = \frac{2\Delta t}{Z_{TL}\Delta l} \left(1 + \frac{Z_{TL}Y_S}{4} \right) \quad (3.45)$$

$$R_d = \frac{\Delta l}{G_S} \quad (3.46)$$

Em um meio com perdas de condutividade σ torna-se,

$$\frac{\partial H_x}{\partial y} + \frac{\partial H_y}{\partial x} = -\varepsilon \frac{\partial E_z}{\partial t} - \sigma E_z \quad (3.47)$$

Assim, a equivalência entre modelo analítico e parâmetros de meio modelado são:

$$\mu \leftrightarrow Z_{TL}\Delta t \quad (3.48)$$

$$\varepsilon \leftrightarrow \frac{2\Delta t}{Z_{TL}\Delta l} \left(1 + \frac{Z_{TL}Y_S}{4} \right) \quad (3.49)$$

$$\sigma \leftrightarrow \frac{G_S}{\Delta l} \quad (3.50)$$

Para o caso com perdas, o processo de implementação do método continua o mesmo, alterando-se somente a matriz de espalhamento, a qual trás informações do meio e acrescentando uma porta para o *stub* na matriz de conexão e transmissão.

4– MÉTODO TRANSMISSION LINE MODELING – JOHN’S SUPER NODE

4.1– INTRODUÇÃO

O método desenvolvido nesta seção baseia-se no TLM-2D apresentado no capítulo 3. Viu-se que o espaço pode ser discretizado por meio de segmentos de linha de transmissão, com impedâncias características determinadas de acordo com os parâmetros do meio em estudo. No entanto, em determinadas situações, a região a ser modelada pode gerar malhas complexas, exigindo assim grande esforço computacional e conseqüentemente uma quantidade de tempo inaceitável para obtenção da solução final.

4.2– MODELAGEM 2D DO MÉTODO TLM-JSN

O método numérico TLM atua no domínio do tempo e pode analisar diversos tipos de estruturas. Estruturas mais complexas requerem maior esforço computacional. A proposta do TLM-JSN é tentar reduzir esse esforço atacando o problema através de três procedimentos. O primeiro propõe a criação de um super nó envolvendo nós adjacentes da malha de modo a resolver a questão do aumento da discretização espacial. A segunda utiliza interpolação de primeira ordem nos passos de tempo para acelerar a obtenção da solução. Já a terceira, baseia-se no uso da redução de ordem de modelo nas matrizes obtidas, diminuindo assim o número de operações realizadas pelo TLM-JSN.

No TLM-2D o cálculo das tensões incidentes e refletidas é dada por (3.32) e (3.33), reescritas aqui de uma forma mais compacta:

$$[V]_k^r = [S][V]_k^i \quad (4.1)$$

$$[V]_{k+1}^i = [C][V]_k^r \quad (4.2)$$

Onde $[S]$ é a matriz de espalhamento e $[C]$ é a matriz de conexão. Esta última descreve a topologia da malha.

O TLM-JSN, por sua vez, utiliza uma formulação de estados para relacionar as tensões incidentes e refletidas. O problema é dividido em dois domínios, o externo e o interno (vide Fig. 4.1). A solução é obtida pelo cálculo da tensão refletida no domínio externo $_{ex}V^r$, causada pelas tensões incidentes do domínio interno $_{in}V^i$ e, pelas tensões incidentes vindas do domínio externo $_{ex}V^i$. Com isso, (4.1) e (4.2) agora podem ser formuladas como a seqüência iterativa:

$$[_{in}V]_k^i = [T^t][_{ex}V]_k^i + [C][S][_{in}V]_{k-1}^i \quad (4.3)$$

$$[_{ex}V]_k^r = [T][S][_{in}V]_k^i \quad (4.4)$$

Onde $[S]$ e $[C]$ são as mesmas matrizes de (4.1) e (4.2). Já a matriz de transmissão $[T]$ representa a relação entre o domínio interno e o domínio externo. A conexão dos domínios interno e externo ocorre nos nós de fronteira do domínio interno [51]. Em resumo, a formulação caracteriza o mesmo que encontrar a matriz de John's, mais detalhes podem ser vistos no artigo muito interessante de Hoefler [52].

A equação (4.3) pode ser compreendida da seguinte maneira. O primeiro termo depois da igualdade representa a contribuição do domínio externo, enquanto o segundo a contribuição do domínio interno para o cálculo da tensão incidente interna no passo de tempo k . A matriz $[T]$ conecta os dois domínios enquanto o produto das matrizes $[C]$ e $[S]$ retrata a propagação interna. Se substituir (4.3) em (4.4), observa-se claramente a dependência das tensões incidentes internas e externas para o cálculo da tensão externa refletida.

A seção 4.3 descreve o procedimento para a obtenção das matrizes relacionando os dois domínios.

4.3– MODELAGEM MATRICIAL

O super nó do TLM-JSN pode ser compreendido com o auxílio da Fig.4.1. Como já descrito, ele é o combinado dos nós adjacentes de modo a formar um grande nó. No

método proposto o espaço continua a ser discretizado da mesma maneira, com a diferença que ao realizar a análise da estrutura em estudo, os nós são considerados constituintes de um bloco maior e não mais isolados. Analisando-se o super nó de forma global, percebe-se um aumento matricial em comparação com a modelagem TLM convencional. No entanto, usando essa formulação é possível escrever o sistema na forma de equações de estados e aplicar técnicas de aceleração.

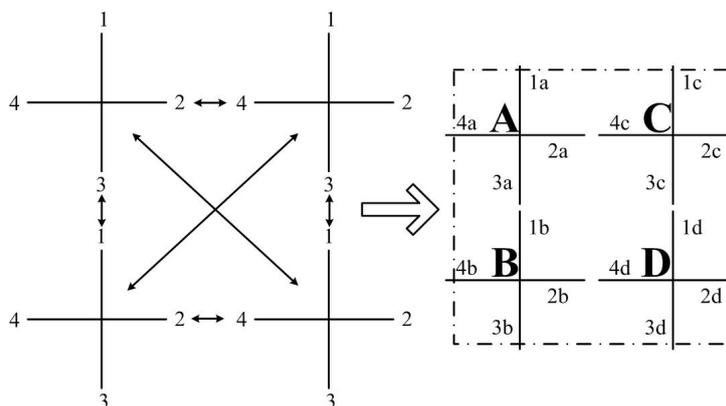


Figura 4.1: Super nó formado pela união de quatro nós adjacentes.

4.3.1– Matriz de Espalhamento

Será demonstrado agora como obter as relações entre as tensões incidentes e refletidas, para o caso particular de um super nó de quatro nós, como mostrado na Fig. 4.2.

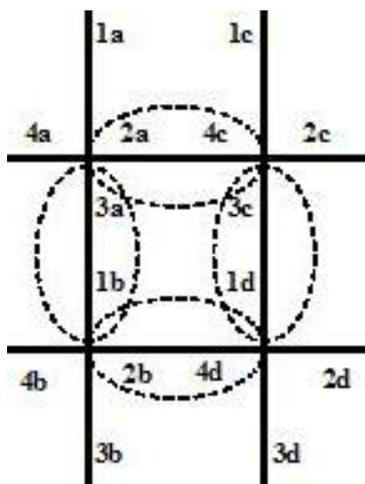


Figura 4.2: Super nó formado por quatro nós adjacentes

Pela Fig. 4.1 observa-se que há nós comuns, ou seja, as portas dos nós individuais estão

conectadas. Esses são os nós: $AB(3a \iff 1b)$, $AC(2a \iff 4c)$, $BD(2b \iff 4d)$ e $CD(1d \iff 3c)$.

Levando em consideração o coeficiente de transmissão dado em (3.25), a tensão total nos nós, no passo de tempo k , pode ser dada por:

$$VA = \frac{1}{2} \sum_{n=1}^4 V_{na}^i \quad (4.5)$$

$$VB = \frac{1}{2} \sum_{n=1}^4 V_{nb}^i \quad (4.6)$$

$$VC = \frac{1}{2} \sum_{n=1}^4 V_{nc}^i \quad (4.7)$$

$$VD = \frac{1}{2} \sum_{n=1}^4 V_{nd}^i \quad (4.8)$$

Tomando as equações (4.5) a (4.8) e isolando os termos de acordo com os ramos internos e externos do esquemático da Fig 4.2, tem-se:

$$VA = \frac{1}{2} (V_{1a}^i + V_{4a}^i) + \frac{1}{2} V_{2a}^i + \frac{1}{2} V_{3a}^i \quad (4.9)$$

$$VB = \frac{1}{2} (V_{3b}^i + V_{4b}^i) + \frac{1}{2} V_{1b}^i + \frac{1}{2} V_{2b}^i \quad (4.10)$$

$$VC = \frac{1}{2} (V_{1c}^i + V_{2c}^i) + \frac{1}{2} V_{3c}^i + \frac{1}{2} V_{4c}^i \quad (4.11)$$

$$VD = \frac{1}{2} (V_{2d}^i + V_{3d}^i) + \frac{1}{2} V_{1d}^i + \frac{1}{2} V_{4d}^i \quad (4.12)$$

Os termos em parênteses das quatro equações anteriores representam as tensões incidentes provenientes dos ramos não comuns do super nó.

As relações para o próximo passo de tempo $(k+1)$ podem ser obtidas através das conexões existentes no esquemático:

$$\begin{cases} V_{3a_{k+1}}^i = V_{1b_k}^r \\ V_{1b_{k+1}}^i = V_{3a_k}^r \end{cases} \quad (4.13)$$

$$\begin{cases} V_{2a_{k+1}}^i = V_{4c_k}^r \\ V_{4c_{k+1}}^i = V_{2a_k}^r \end{cases} \quad (4.14)$$

$$\begin{cases} V_{2b_{k+2}}^i = V_{4d_{k+1}}^r \\ V_{4d_{k+2}}^i = V_{2b_{k+1}}^r \end{cases} \quad (4.15)$$

$$\begin{cases} V_{3c_{k+2}}^i = V_{4d_{k+1}}^r \\ V_{4d_{k+2}}^i = V_{3c_{k+1}}^r \end{cases} \quad (4.16)$$

Como a tensão refletida é igual a tensão total menos a tensão incidente, tem-se:

$$\begin{cases} V_{1b_k}^r = VB - V_{1b_k}^i \\ V_{3a_k}^r = VA - V_{3a_k}^i \end{cases} \quad (4.17)$$

$$\begin{cases} V_{2a_k}^r = VA - V_{2a_k}^i \\ V_{4c_k}^r = VC - V_{4c_k}^i \end{cases} \quad (4.18)$$

$$\begin{cases} V_{2b_k}^r = VB - V_{2b_k}^i \\ V_{4d_k}^r = VD - V_{4d_k}^i \end{cases} \quad (4.19)$$

$$\begin{cases} V_{3c_k}^r = VC - V_{3c_k}^i \\ V_{1d_k}^r = VD - V_{1d_k}^i \end{cases} \quad (4.20)$$

Tomando as equações (4.9)-(4.12) e substituindo em (4.17)-(4.20), obtêm-se:

$$V_{1b_k}^r = \frac{1}{2} (V_{3b_k}^i + V_{4b_k}^i) - \frac{1}{2} V_{1b_k}^i + \frac{1}{2} V_{2b_k}^i \quad (4.21)$$

$$V_{3a_k}^r = \frac{1}{2} (V_{1a_k}^i + V_{4a_k}^i) + \frac{1}{2} V_{2a_k}^i - \frac{1}{2} V_{3a_k}^i \quad (4.22)$$

$$V_{2a_k}^r = \frac{1}{2} (V_{1a_k}^i + V_{4a_k}^i) - \frac{1}{2} V_{2a_k}^i + \frac{1}{2} V_{3a_k}^i \quad (4.23)$$

$$V_{4c_k}^r = \frac{1}{2} (V_{1c_k}^i + V_{2c_k}^i) + \frac{1}{2} V_{3c_k}^i - \frac{1}{2} V_{4c_k}^i \quad (4.24)$$

$$V_{2b_k}^r = \frac{1}{2} (V_{3b_k}^i + V_{4b_k}^i) - \frac{1}{2} V_{2b_k}^i + \frac{1}{2} V_{1b_k}^i \quad (4.25)$$

$$V_{4d_k}^r = \frac{1}{2} (V_{2d_k}^i + V_{3d_k}^i) + \frac{1}{2} V_{1d_k}^i - \frac{1}{2} V_{4d_k}^i \quad (4.26)$$

$$V_{3c_k}^r = \frac{1}{2} (V_{1c_k}^i + V_{2c_k}^i) - \frac{1}{2} V_{3c_k}^i + \frac{1}{2} V_{4c_k}^i \quad (4.27)$$

$$V_{1d_k}^r = \frac{1}{2} (V_{2d_k}^i + V_{3d_k}^i) - \frac{1}{2} V_{1d_k}^i + \frac{1}{2} V_{4d_k}^i \quad (4.28)$$

Seguindo o mesmo procedimento das equações (4.21) a (4.28) obtêm-se as tensões refletidas dos demais ramos. Assim, a matriz $[S]$ global para quatro nós é escrita da seguinte forma:

$$\begin{pmatrix} V_{1a} \\ V_{2a} \\ V_{3a} \\ V_{4a} \\ V_{1b} \\ V_{2b} \\ V_{3b} \\ V_{4b} \\ V_{1c} \\ V_{2c} \\ V_{3c} \\ V_{4c} \\ V_{1d} \\ V_{2d} \\ V_{3d} \\ V_{4d} \end{pmatrix}^r = \frac{1}{2} \begin{pmatrix} -1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} V_{1a} \\ V_{2a} \\ V_{3a} \\ V_{4a} \\ V_{1b} \\ V_{2b} \\ V_{3b} \\ V_{4b} \\ V_{1c} \\ V_{2c} \\ V_{3c} \\ V_{4c} \\ V_{1d} \\ V_{2d} \\ V_{3d} \\ V_{4d} \end{pmatrix}^i$$

Reescrevendo essa mesma matriz, agora separando as tensões refletidas de acordo com os ramos internos e externos do super nó, obtêm-se:

$$\begin{pmatrix} V_{1a} \\ V_{4a} \\ V_{3b} \\ V_{4b} \\ V_{1c} \\ V_{2c} \\ V_{2d} \\ V_{3d} \\ V_{2a} \\ V_{3a} \\ V_{1b} \\ V_{2b} \\ V_{3c} \\ V_{4c} \\ V_{1d} \\ V_{4d} \end{pmatrix}^r = \frac{1}{2} \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} V_{1a} \\ V_{4a} \\ V_{3b} \\ V_{4b} \\ V_{1c} \\ V_{2c} \\ V_{2d} \\ V_{3d} \\ V_{2a} \\ V_{3a} \\ V_{1b} \\ V_{2b} \\ V_{3c} \\ V_{4c} \\ V_{1d} \\ V_{4d} \end{pmatrix}^i$$

Observando a matriz anterior pode-se reescrever a matriz $[S]$ como:

$$\begin{bmatrix} S \end{bmatrix} = \begin{bmatrix} S_{ee} & S_{ei} \\ S_{ie} & S_{ii} \end{bmatrix} \quad (4.29)$$

Onde S_{ee} é a relação entre as tensões refletidas e incidentes dos ramos externos, S_{ei} entre as tensões refletidas dos ramos internos e tensões incidentes dos ramos externos, S_{ie} entre as tensões refletidas dos ramos externos e tensões incidentes dos ramos internos e por fim, S_{ii} entre as tensões refletidas e incidentes dos ramos internos do super nó.

Pode-se dizer que a malha do super nó analisado possuía $(2, 2)$ nós. Essa é uma maneira de representação similar à formulação matricial. Ou seja, há dois nós por linha e dois por coluna, num total de quatro nós. A técnica empregada para a obtenção da matriz $[S]$ para esse caso específico pode ser expandida para qualquer super nó agrupado em (M, N) nós. Onde o par (M, N) representa a posição de um nó que se encontra na linha M e na coluna N da malha. Dessa forma a matriz $[S]$ será quadrada de ordem igual a $(4 \times M \times N)$.

À medida que o número de nós reunidos no super nó aumenta, a matriz $[S]$ torna-se consideravelmente grande. Para contornar tal problema decompõe-se a matriz $[S]$ por meio do Produto Tensorial de Kronecker [53], facilitando assim sua construção da mesma e reduzindo o tempo gasto no processo de discretização das estruturas. Com isso a matriz pode ser escrita como:

$$S = I_2 \otimes I_{2n} \otimes K + P_2 \otimes I_2 \otimes U \quad (4.30)$$

Onde (\otimes) é o produto tensorial de Kronecker, I_2 é a matriz identidade de ordem dois, I_{2n} é a matriz identidade de ordem igual ao número de nós, P_2 é a inversão binária da matriz I_2 , K e U . De uma forma mais específica:

$$I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (4.31)$$

$$P_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (4.32)$$

$$I_{2n} = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}_{n \times n} \quad (4.33)$$

$$K = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \quad (4.34)$$

$$U = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad (4.35)$$

O procedimento para a obtenção da equação (4.30) encontra-se no Apêndice.

4.3.2– Matriz de Conectividade

Como já exposto, a matriz $[C]$ descreve a topologia da malha. Ela mostra de forma concisa e direta as conexões existentes entre os ramos dos nós individuais. O procedimento para sua obtenção pode ser entendido com o auxílio da Fig. 4.3, para o caso particular de quatro nós.

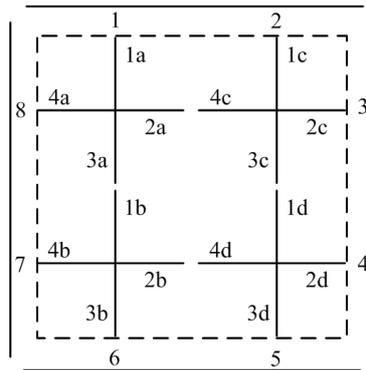


Figura 4.3: Obtenção das Matrizes $[C]$ e $[T]$ para o super nó de quatro nós.

A matriz $[C]$ é quadrada de ordem $4 \times M \times N$. Para o super nó em estudo de $(2, 2)$ nós, a matriz é quadrada de ordem 16. Observa-se que se o nó A é conectado ao nó C por meio do ramo $(2a - 4c)$. Assim as posições correspondentes aos pares $(2a, 4c)$ e $(4c, 2a)$ na matriz C são preenchidas com um elemento unitário. Generalizando para os demais nós e suas interconexões obtém-se:

	1a	2a	3a	4a	1b	2b	3b	4b	1c	2c	3c	4c	1d	2d	3d	4d
1a																
2a												1				
3a					1											
4a																
1b			1													
2b																1
3b																
4b																
1c																
2c																
3c													1			
4c		1														
1d												1				
2d																
3d																
4d							1									

Nota-se que as linhas e colunas correspondem aos ramos de cada nó na malha e que também a matriz é simétrica. O procedimento para se obter a matriz C para um número qualquer de nós pode ser sumarizado da seguinte forma: se um ramo m do nó X está conectado a um ramo n do nó Y , a posição correspondente a (m, n) nessa matriz receberá o valor unitário, assim como também a posição (n, m) .

4.3.3– Matriz de Transmissão

A matriz $[T]$, por sua vez, define as conexões existentes entre os nós do domínio interno e domínio externo. Essa matriz possui $G(4 \times M \times N)$ elementos, onde G é o número de portas externas da malha.

A obtenção da matriz $[T]$ é semelhante a da matriz $[C]$ e é realizada da seguinte maneira: se uma porta g está conectada a um ramo m do nó X , o elemento (g, m) da matriz $[T]$ terá valor unitário. O mesmo não vale para o elemento (m, g) , se este existir, pois a matriz não é simétrica.

Tomando a Fig. 4.3 como exemplo, tem-se que a matriz $[T]$ é dada por:

	1a	2a	3a	4a	1b	2b	3b	4b	1c	2c	3c	4c	1d	2d	3d	4d
1	1															
2									1							
3										1						
4													1			
5																1
6							1									
7								1								
8				1												

Na equação (4.36) as linhas correspondem às portas externas e as colunas aos ramos dos

nós na malha. O número de elementos não-nulos dessa matriz é igual ao número de portas da malha. No caso acima o número de portas (elementos unitários) é igual a 8. O sistema de equações em (4.3) e (4.4) para o super nó estudado pode ser reescrito de acordo com as dimensões das matrizes, na forma:

$$\{[in V]_k^i\}_{16 \times 1} = \left\{ [T^t]_{16 \times 8} \{[ex V]_k^i\}_{8 \times 1} \right\}_{16 \times 1} + \left\{ \{[C][S]\}_{16 \times 16} \{[in V]_{k-1}^i\}_{16 \times 1} \right\}_{16 \times 1} \quad (4.36)$$

$$\{[ex V]_k^r\}_{16 \times 1} = \{[T][S]\}_{16 \times 16} \{[in V]_k^i\}_{16 \times 1} \quad (4.37)$$

4.4– INTERPOLAÇÃO LINEAR DAS EQUAÇÕES DE ESTADO DO TLM-JSN

O algoritmo de aceleração aqui descrito baseia-se na forma a qual (4.3) é escrita. Após algumas manipulações algébricas pode-se aplicar a interpolação de primeira ordem. Sejam as tensões incidentes calculadas desde o passo de tempo $k - p$ até k :

$$\begin{aligned} [in V]_k^i &= [T^t][ex V]_k^i + [C][S][in V]_{k-1}^i \\ \{[C][S]\} [in V]_{k-1}^i &= \{[C][S]\} [T^t][ex V]_{k-1}^i + \{[C][S]\}^2 [in V]_{k-2}^i \\ \{[C][S]\}^2 [in V]_{k-2}^i &= \{[C][S]\}^2 [T^t][ex V]_{k-2}^i + \{[C][S]\}^3 [in V]_{k-3}^i \\ \{[C][S]\}^3 [in V]_{k-3}^i &= \{[C][S]\}^3 [T^t][ex V]_{k-3}^i + \{[C][S]\}^4 [in V]_{k-4}^i \\ &\vdots \\ \{[C][S]\}^{p-1} [in V]_{k-p-1}^i &= \{[C][S]\}^{p-1} [T^t][ex V]_{k-p-1}^i + \{[C][S]\}^p [in V]_{k-p}^i \end{aligned} \quad (4.38)$$

Somando as equações acima e eliminando os termos iguais em ambos lados das equações, chega-se a:

$$[in V]_k^i = \sum_{n=0}^{p-1} \{[C][S]\}^n [T^t][ex V]_{k-n}^i + \{[C][S]\}^p [in V]_{k-p}^i \quad (4.39)$$

A razão p em (4.39) representa os passos de tempo que serão omitidos. Nota-se também que a equação é exata em relação a expansão do sistema em forma de somatório, e não oferece nenhuma vantagem do ponto de vista computacional. Entretanto, pode-se calcular $[ex V]_{k-n}^i$ em função de $[ex V]_k^i$ e $[ex V]_{k-p}^i$ desde que os sinais de entrada variem lentamente no tempo. Isto se aplica para certas excitações Gaussianas e considerando a dualidade

entre frequência e tempo, quanto maior o intervalo de tempo analisado menor será a largura de banda no domínio da frequência.

Seja o somatório da equação (4.39) na forma expandida:

$$\begin{aligned} \sum_{n=0}^{p-1} \{[C][S]\}^n [T^t][exV]_{k-n}^i &= [I][T^t][exV]_k^i + \{[C][S]\} [T^t][exV]_{k-1}^i + \\ &+ \{[C][S]\}^2 [T^t][exV]_{k-2}^i + \{[C][S]\}^3 [T^t][exV]_{k-3}^i + \\ &+ \dots + \{[C][S]\}^{p-1} [T^t][exV]_{k-p-1}^i \end{aligned} \quad (4.40)$$

Onde $[I]$ é a matriz identidade. Agora será realizada a interpolação de primeira ordem da tensão incidente externa entre os instantes k e $k - p$. As tensões $[exV]_{k-n}^i$ são calculadas por meio de:

$$\begin{aligned} [exV]_{k-p-1}^i &= \left\{ \left(\frac{1}{p} \right) [exV]_k^i + \left(\frac{p-1}{p} \right) [inV]_{k-p}^i \right\} \\ &\quad \vdots \\ [exV]_{k-n}^i &= \left\{ \left(\frac{p-n}{p} \right) [exV]_k^i + \left(\frac{n}{p} \right) [inV]_{k-p}^i \right\} \\ &\quad \vdots \\ [exV]_{k-3}^i &= \left\{ \left(\frac{p-3}{p} \right) [exV]_k^i + \left(\frac{3}{p} \right) [inV]_{k-p}^i \right\} \\ [exV]_{k-2}^i &= \left\{ \left(\frac{p-2}{p} \right) [exV]_k^i + \left(\frac{2}{p} \right) [inV]_{k-p}^i \right\} \\ [exV]_{k-1}^i &= \left\{ \left(\frac{p-1}{p} \right) [exV]_k^i + \left(\frac{1}{p} \right) [inV]_{k-p}^i \right\} \end{aligned} \quad (4.41)$$

Substituindo as equações de (4.40) e (4.41) em (4.39) e fazendo as manipulações necessárias obtém-se:

$$[inV]_k^i = [A][exV]_k^i + [B][exV]_{k-p}^i + ([C][S])^p [inV]_{k-p}^i \quad (4.42)$$

Onde as matrizes $[A]$ e $[B]$ são calculadas através de:

$$[A] = \left\{ [I] + \sum_{n=1}^{p-1} \left(1 - \frac{n}{p} \right) ([C][S])^n \right\} [T] \quad (4.43)$$

$$[B] = \left\{ \sum_{n=1}^{p-1} \left(\frac{n}{p} \right) ([C][S])^n \right\} [T] \quad (4.44)$$

Dessa maneira, a tensão num nó interno em um passo de tempo k , $[inV]_k^i$ é calculada utilizando a combinação linear da tensão incidente externa entre os passos tempo k e $k - p$. Ou seja, há um salto de p passos de tempo para o cálculo da tensão atual.

Analisando a equação (4.42) observa-se que a tensão interna incidente no passo de tempo k também depende da própria tensão interna incidente no passo de tempo anterior $k - p$. Esse termo representa a contribuição da propagação interna na malha para o cálculo da tensão incidente em p passos de tempo à frente.

Já a aceleração do algoritmo só é possível controlando-se as tensões incidentes externas. Isso se deve ao fato de que os parâmetros de qualquer estrutura discretizada pelo método TLM são fixos, como dadas por (3.22) e (3.23), e conseqüentemente o tempo de propagação interno Δt também o é. Daí a necessidade de se trabalhar com excitações que variem lentamente com o tempo para que haja ganho de processamento. Por outro lado, isso acarreta a perda de informação de altas frequências. Sendo assim, esse método é mais indicado para análise de circuitos integrados (CI's) [51].

Será visto que os ganhos de processamento são mais acentuados para valores de p acima de 150. Grandes valores de p significam sinais de entrada passa-baixas de bandas estreitas. Para que o método desenvolvido funcione adequadamente a banda do sinal de entrada deve ser menor que $f_{max} = \frac{1}{2p\Delta t}$.

4.5– REDUÇÃO DE ORDEM DO MODELO

Como foi visto, a obtenção das matrizes é realizada em função da discretização. As mesmas podem se tornar extremamente grandes, ocasionando um esforço computacional muito maior para a sua criação e, por conseguinte, para a resolução do sistema linear em (4.3) e (4.5). Além disso, há um grande dispêndio de tempo e memória para a resolução da interpolação de primeira ordem em (4.42).

A matriz resultante da multiplicação entre a matriz de conexão $[C]$ e da matriz de espalhamento $[S]$, em (4.3) e em (4.42), é uma das principais responsáveis pela grande quantidade de cálculos a serem realizados. Para efeito de análise, o produto demanda $(4 \times M \times N)^3$ operações de multiplicação, onde $4 \times M \times N$ é a ordem da matriz $[C]$ e também da matriz $[S]$.

Com isso, qualquer aumento linear no número de nós gera um triplo acréscimo no número de operações de multiplicação, apesar da esparsidade das matrizes resolver parte desse problema.

Nesta seção serão apresentadas duas soluções para estes problemas através da redução de ordem de modelo. Os dois algoritmos são chamados de ROM-1 e ROM-2.

4.5.1– Primeira Redução de Modelo - ROM 1

O algoritmo aqui apresentado tem como base a decomposição de Schur para efetuar a redução de modelo [54]. A decomposição de Schur transforma uma matriz X quadrada em uma matriz Q quasitriangular e uma matriz ortogonal U . Então:

$$X = UQU^T \quad (4.45)$$

Onde $U^T U = I$ e I é a matriz identidade com as mesmas dimensões de U .

Tomando as equações do sistema linear em (4.3) e (4.4) e reescrevendo-as da seguinte maneira, tem-se:

$$w_k = Dx_k + Ew_{k-1} \quad (4.46)$$

$$y_k = Fw_k \quad (4.47)$$

Onde $w_k = [inV]_k^i$, $x_k = [exV]_k^i$, $y_k = [exV]_k^r$, $D = [T^t]$, $E = [C][S]$ e $F = [T][S]$. Aplicando a decomposição de Schur em (4.46) e (4.47), demonstra-se que o sistema linear pode ser reduzido como segue abaixo:

$$g_k = UDx_k + Qg_{k-1} \quad (4.48)$$

$$y_k = FQ^T g_k \quad (4.49)$$

Onde $g = Tw$ e $w = T^T g$. Após a aplicação da redução de modelo, o ganho no sistema do TLM-JSN é de aproximadamente 40%. Esta redução não só diminui as operações como reduz o esforço computacional economizando ainda mais memória.

4.5.2– Segunda Redução de Modelo - ROM 2

O segundo algoritmo é baseado na redução de ordem de modelo descrita em [55]. Essa redução utiliza a transformada Z , o que traz algumas vantagens que serão descritas nessa seção. Partindo do sistema linear (4.3) e (4.4), e reescrevendo-o como:

$$w_k = Ex_k + Fw_{k-1} \quad (4.50)$$

$$y_k = Gw_k \quad (4.51)$$

Onde novamente $w_k = [{}_{in}V]_k^i$, $x_k = [{}_{ex}V]_k^i$, $y_k = [{}_{ex}V]_k^r$ e $E = [T^t]$, $F = [C][S]$ e $G = [T][S]$.

Substituindo as equações do sistema em (4.50) e (4.51), tem-se:

$$w_k = Ex_k + Fw_{k-1} \quad (4.52)$$

$$y_k = GEx_k + GFw_{k-1} \quad (4.53)$$

A razão para essas substituições em (4.52) e (4.53) surge pela facilidade de se aplicar a transformada Z :

$$zW(z) = zEX(z) + FW(z) \quad (4.54)$$

$$zY(z) = zGEX(z) + GFW(z) \quad (4.55)$$

Escrevendo a matriz F como:

$$F = R^{-1}\Lambda R \quad (4.56)$$

Onde Λ é a matriz diagonal dos autovalores de F e RR^{-1} . Substituindo (4.56) em (4.54) e (4.55), fazendo $V = RW(z)$ e rearranjando obtém-se:

$$zV = zREX(z) + \Lambda V \quad (4.57)$$

$$zY(z) = zGEX(z) + GR^{-1}\Lambda V \quad (4.58)$$

É em (4.57) e (4.58) que se dá a redução de modelo. Ela é possível controlando-se o posto (quantidade de linhas ou colunas linearmente independentes) da matriz Λ . Como Λ é

uma matriz diagonal a eliminação de autovalores específicos é que garante a redução da ordem de modelo. Isso pode ser entendido com ajuda da matriz abaixo.

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & \dots & 0 \\ 0 & \lambda_2 & 0 & \dots & 0 \\ 0 & 0 & \lambda_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \lambda_j \end{bmatrix}_{j \times j} \Rightarrow \Lambda' = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \lambda_i \end{bmatrix}_{i \times i} \quad (4.59)$$

Pela equação observa-se que $j > i$. Pode se mostrar que os autovalores eliminados representam as altas frequências enquanto os autovalores remanescentes as baixas frequências.

Após o procedimento, utiliza-se a transformada Z inversa para retornar ao domínio temporal. Procede-se assim ao cálculo iterativo das tensões incidentes internas e tensões refletidas externas, agora com as matrizes reduzidas. Será visto que a performance do ROM-2 é superior à do ROM-1, na análise de diversas estruturas.

4.6– FLUXOGRAMA DO MÉTODO TLM-JSN

A Fig. 4.4 mostra o fluxograma do Método TLM-JSN para a simulação de uma estrutura qualquer. O método pode ser resumido da seguinte forma. Entra-se com a topologia da estrutura a ser analisada e suas dimensões. Em seguida segue-se para a fase de preparação, onde é realizado o cálculo das matrizes de espalhamento $[S]$, de conexão $[C]$ e de transmissão $[T]$. Com as matrizes calculadas passa-se para a próxima fase. É nela que se definem: o número de passos de tempo ($NT = k$) para se obter a solução final; as portas de entrada e de saída, onde será inserida a excitação e onde será extraída a tensão refletida, respectivamente; as tensões incidentes externas e internas, esta última sendo representada por um vetor nulo de modo a limpar as possíveis tensões internas pré-existentes; a quantidade de saltos de tempo expresso pela razão p . Pode-se dizer que a inicialização é também uma fase onde se entra com parâmetros (dados de entrada), mas estes estão vinculados diretamente à estrutura analisada. Com tudo definido corretamente, faz-se uso da redução de modelo para diminuir o número de cálculos do algoritmo propriamente dito. Utiliza-se um dos dois procedimentos: ROM-1 ou ROM-2. Considera-se que os passos anteriores constituem a preparação completa do método. Finalmente, chega-se à

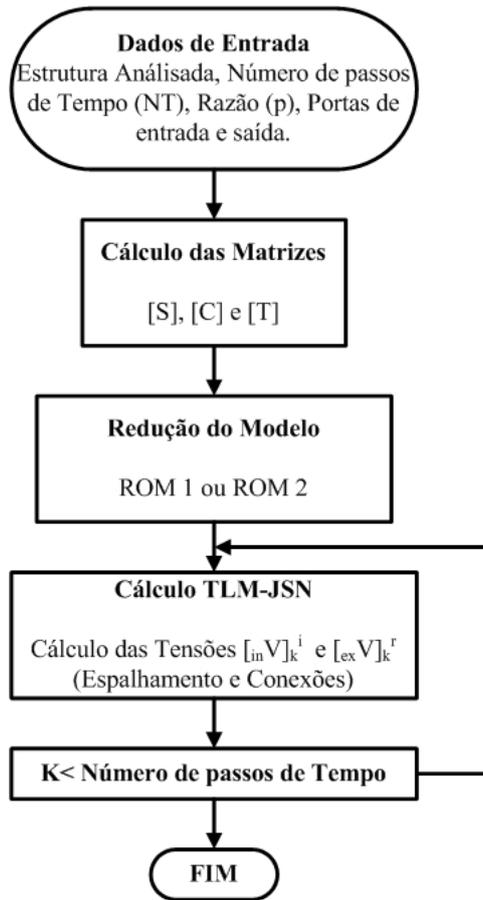


Figura 4.4: Fluxograma do algoritmo TLM-JSN.

fase do algoritmo TLM-JSN, onde em NT passos de tempo e utilizando-se uma razão p são calculadas iterativamente as tensões incidentes internas e as tensões refletidas externas conforme as equações (4.3) e (4.4). Como resultado, ao final do método, obtêm-se a tensão desejada ${}_{ex}V^r$, na porta de saída especificada.

5– APLICAÇÃO DA ENVOLTÓRIA COMPLEXA NO MÉTODO TLM-JSN

5.1– INTRODUÇÃO

Diversos métodos já utilizaram envoltória complexa para simulação de guias de onda e circuitos, os mais conhecidos são o EFVE e ADI-FDTD [57–61]. Neste capítulo mostram-se duas formas de aplicação da envoltória complexa na modelagem numérica utilizando o método TLM-JSN [51, 56]. A primeira consiste em chegar ao mesmo sistema descrito no capítulo 4 nas equações (4.3) e (4.4). No entanto essa técnica possui uma característica de envelope do conteúdo em frequência. Além disso, o equacionamento no formato de equações de estados pode trazer benefício significativo no que se refere a aplicação da interpolação linear de primeira ordem na modelagem da envoltória complexa. Já na segunda técnica é obtida usando o mesmo procedimento, porém mais simplificada e utilizando interpolação de primeira ordem para obter um controle da ordem do sistema.

Novas técnicas para diminuição do esforço computacional na resolução dos sistemas são apresentadas nas seções seguintes, apresentando a teoria que descreve a aplicação da envoltória no TLM e TLM-JSN.

5.2– APLICAÇÃO DA ENVOLTÓRIA COMPLEXA EXATA NO MÉTODO TLM-JSN

O objetivo desta seção é calcular a envoltória exata para o TLM-JSN. Para chegar na fórmula exata (sem pulos de passos de tempo) é necessário rever alguns pontos. Partindo

do sistema matricial descrito em [51]:

$$\begin{aligned} [W]^r &= [T_r] [V]^i + z^{-1} [M] [W]^r \\ [V]^r &= [T_i] [W]^r \end{aligned} \quad (5.1)$$

Onde $[W]^r = [in V]_k^i$, $[T_r] = [ex V]_k^i$, $[V]^r = [ex V]_k^r$ e $[T_r] = [T^t]$, $[M] = [C][S]$ e $[T_i] = [T][S]$.
Pode-se reescrever a equação (5.1) como:

$$\{[U] - z^{-1} [M]\} [W]^r = [T_r] [V]^i \quad (5.2)$$

Chamando:

$$[A(z)] = ([U] - z^{-1} [M])^{-1} \quad (5.3)$$

Esta equação pode ser escrita na forma expandida da seguinte maneira:

$$\begin{aligned} [A(z)] &= ([U] - z^{-1} [M])^{-1} \\ &= [U] + z^{-1} [M] + z^{-2} [M]^2 + \dots + z^{-(k+1)} [M]^{k+1} + \dots \end{aligned} \quad (5.4)$$

Utilizando esta recursividade pode-se escrever em uma forma compacta:

$$\begin{aligned} [A(z)] &= [U] + z^{-1} [M]^{-1} ([U] - z^{-1} [M])^{-1} \\ &= [U] + z^{-1} [M] [A(z)] \end{aligned} \quad (5.5)$$

Considerando-se uma frequência em torno de uma frequência central, no domínio Z isto pode ser escrito como:

$$z_0^{-1} = z^{-1} z_c^{-1} \quad (5.6)$$

Assim obtém-se a equação:

$$\begin{aligned} [A(z_0)] &= [U] + z_0^{-1} [M]^{-1} ([U] - z_0^{-1} [M])^{-1} \\ &= [U] + z_0^{-1} [M] [A(z)] \end{aligned} \quad (5.7)$$

De modo muito útil pode-se expandir $A(z_0)$ em sua forma de Taylor para z tendendo a 1, ou seja, na frequência central, a série exata pode ser expressa de forma recursiva.

No entanto, isto fica mais simples observando um exemplo algébrico. Para este caso a expansão se torna:

$$\begin{aligned} a &= \frac{1}{1 - zz_cm} \\ &= \frac{1}{1 - z_cm} - \frac{z_cm(z-1)}{1 - z_cm} \frac{1}{1 - z_cm} + \left(\frac{z_cm(z-1)}{1 - z_cm} \right)^2 \frac{1}{1 - z_cm} + \dots \end{aligned} \quad (5.8)$$

Ora isto é recursivo, pois pode ser escrito:

$$a = \frac{1}{1 - z_cm} - \frac{z_cm(z-1)}{1 - z_cm} \times \left[\frac{1}{1 - z_cm} - \frac{z_cm(z-1)}{1 - z_cm} \frac{1}{1 - z_cm} + \dots \right] \quad (5.9)$$

Utilizando a recursividade obtem-se a expressão:

$$\begin{aligned} a &= \frac{1}{1 - zz_cm} \\ &= \frac{1}{1 - z_cm} - \frac{z_cm(z-1)}{1 - z_cm} a \end{aligned} \quad (5.10)$$

Isto pode ser escrito de forma mais simples resultando em:

$$a = \frac{1}{1 - z_cm} - \frac{z_czm}{1 - z_cm} a + \frac{z_cm}{1 - z_cm} a \Leftrightarrow a - \frac{z_cm}{1 - z_cm} a = \frac{1}{1 - z_cm} - \frac{z_czm}{1 - z_cm} a \quad (5.11)$$

Por fim:

$$a = \frac{1}{1 - z_cm} - \frac{z_czm}{1 - z_cm} a \quad (5.12)$$

No caso de expressões matriciais, a série de Taylor continua valendo, portanto tem-se a expressão recursiva:

$$\begin{aligned} [A(z_0)] &= ([U] - z^{-1}z_c^{-1}[M])^{-1} \\ &= ([U] - z_c^{-1}[M])^{-1} [U] - (z^{-1} - 1) z_c^{-1} \times \\ &\times [M] ([U] - z_c^{-1}[M])^{-1} [A(z_0)] \end{aligned} \quad (5.13)$$

No entanto, do mesmo modo que o anterior, esta formulação pode ser compactada. Chamando:

$$\begin{aligned} [A(z_c)] &= ([U] - z_c^{-1}[M])^{-1} \\ &= [U] + z_c^{-1}[M] [A(z_c)] \end{aligned} \quad (5.14)$$

E simplificando a expressão:

$$[A(z_0)] = [A(z_c)] [U] - (z^{-1} - 1) z_c^{-1} \times [M] [A(z_c)] [A(z_0)] \quad (5.15)$$

E agora expandindo obtem-se:

$$[A(z_0)] = [A(z_c)] [U] - z^{-1} z_c^{-1} [M] [A(z_c)] [A(z_0)] + z_c^{-1} [M] [A(z_c)] [A(z_0)] \quad (5.16)$$

Rearranjando, chega-se a:

$$[A(z_0)] - z_c^{-1} [M] [A(z_c)] [A(z_0)] = [A(z_c)] [U] - z^{-1} z_c^{-1} [M] [A(z_c)] [A(z_0)] \quad (5.17)$$

Colocando termos em evidência consegue-se:

$$([U] - z_c^{-1} [M] [A(z_c)]) [A(z_0)] = [A(z_c)] [U] - z^{-1} z_c^{-1} [M] [A(z_c)] [A(z_0)] \quad (5.18)$$

Portanto, pode-se escrever a nova expressão:

$$\begin{aligned} [A(z_0)] &= ([U] - z_c^{-1} [M] [A(z_c)])^{-1} [A(z_c)] [U] - z^{-1} z_c^{-1} ([U] - z_c^{-1} [M] [A(z_c)])^{-1} \times \\ &\times [M] [A(z_c)] [A(z_0)] \end{aligned} \quad (5.19)$$

Na formulação recursiva:

$$\begin{aligned} [W]^r &= ([U] - z_c^{-1} [M] [A(z_c)])^{-1} [A(z_c)] [T_r] [V]^i + z^{-1} z_c^{-1} ([U] - z_c^{-1} [M] [A(z_c)])^{-1} \times \\ &\times [M] [A(z_c)] [W]^r \end{aligned} \quad (5.20)$$

$$[V]^r = [T_i] [W]^r \quad (5.21)$$

Chamando:

$$[D(z_c)] = ([U] - z_c^{-1} [M] [A(z_c)])^{-1} \times [A(z_c)] [T_r] \quad (5.22)$$

$$[E(z_c)] = z_c^{-1} ([U] - z_c^{-1} [M] [A(z_c)])^{-1} \times [M] [A(z_c)] \quad (5.23)$$

$$[F(z_c)] = [T_i] \quad (5.24)$$

Assim, consegue-se chegar a formulação final da envoltória:

$$[W]^r = [D(z_c)] [V]^i + z^{-1} [E(z_c)] [W]^r \quad (5.25)$$

$$[V]^r = [F(z_c)] [W]^r \quad (5.26)$$

Nota-se que o sistema obtido está na forma mais apropriada na utilização de pulos no tempo. Na seção seguinte apresenta-se a formulação com algumas aproximações matemáticas para obter matrizes menores, assim condicionando o sistema para uma aproximação mais refinada em frequência.

5.3– APLICAÇÃO DA ENVOLTÓRIA COMPACTA NO TLM-JSN

Agora mostra-se a implementação da envoltória simplificada, sem pulos no tempo. No entanto esta técnica utiliza interpolação linear para obter uma melhor aproximação para o problema em estudo. Esta técnica resolve parcialmente o problema das matrizes extremamente grandes diminuindo a alocação de memória em cada simulação. Ao invés de guardar três matrizes, duas de $4 \times (N1 + N2 + 4)$ por $4 \times N1 \times N2$ e uma matriz de tamanho igual a $4 \times N1 \times N2$ por $4 \times N1 \times N2$, esta técnica com envoltória guarda três matrizes de $4 \times (N1 + N2 + 4)$ por $4 \times (N1 + N2 + 4)$. Mas, como todas as outras técnicas, existem desvantagens. A principal desvantagem é que essa técnica só é precisa com conteúdo de frequência na vizinhança da envoltória devido a interpolação utilizada. Partindo do sistema transformado no domínio Z , tem-se a seguinte expressão genérica:

$$[W]^r = [T_r][V]^i + z^{-1}[M][W]^r \quad (5.27)$$

$$[V]^r = [T_i][W]^r \quad (5.28)$$

Onde $[W]^r = [in V]_k^i$, $[T_r] = [ex V]_k^i$, $[V]^r = [ex V]_k^r$ e $[T_r] = [T^t]$, $[M] = [C][S]$ e $[T_i] = [T][S]$. Pode-se simplificar a expressão (5.28) considerando algumas operações simples em (5.27) e substituindo em (5.28):

$$[V]^r = [T_i]([U] - z^{-1}[M])^{-1}[T_r][V]^i \quad (5.29)$$

Chamando esta expressão de:

$$\begin{aligned} [A(z)] &= ([U] - z^{-1}[M])^{-1} \\ &= [U] + z^{-1}[M][A(z)] \end{aligned} \quad (5.30)$$

Caso o interesse seja na resposta em uma vizinhança específica de uma frequência deter-

minada pode-se fazer:

$$z_0^{-1} = z^{-1} z_c^{-1} \quad (5.31)$$

Assim, a expressão de modo genérico pode ser representada, considerando a resposta em uma vizinhança específica de uma frequência determinada, de modo a obter:

$$\begin{aligned} [A(z_0)] &= [U] + z_0^{-1} [M] [A(z_0)] \\ z_0^{-1} [M] [A(z_0)] &= z_0^{-1} [M] [U] + z_0^{-2} [M]^2 [A(z_0)] \\ &\dots \\ z_0^{-(p-1)} [M] [A(z_0)] &= z_0^{-(p-1)} [M]^{(p-1)} [U] + z_0^{-p} [M]^p [A(z_0)] \end{aligned} \quad (5.32)$$

Somando os termos:

$$[A(z_0)] = \sum_{k=0}^{p-1} z_0^{-k} [M]^k [U] + z_0^{-p} [M]^p [A(z_0)] \quad (5.33)$$

Esta expressão leva em consideração o problema do pulo de tempo que é de grande interesse. Mas nota-se que ela não resolve o problema, pois a princípio obtêm-se a mesma expressão devido a recursividade. Para contornar isto utiliza-se a interpolação linear:

$$([U] - z^{-p} [M]^p) ([U] - z^{-1} [M])^{-1} = \sum_{n=0}^{p-1} z^{-n} [M]^n \quad (5.34)$$

$$\sum_{n=0}^{p-1} z^{-n} [M]^n = [A] + z^{-p} [B] \quad (5.35)$$

Onde:

$$\begin{aligned} [A] &= \left\{ [U] + \sum_{n=1}^{p-1} \left(1 - \frac{n}{p}\right) [M]^n \right\} \\ [B] &= \left\{ \sum_{n=1}^{p-1} \frac{n}{p} \cdot [M]^n \right\} \end{aligned} \quad (5.36)$$

Esta expressão pode ser simplificada pelo fato de estar calculando a envoltória. Deste modo, pode-se escrever:

$$[A] = [A(z_c)] - \frac{1}{p} z_c^{-1} [M] [A(z_c)]^2 ([U] - z_c^{-p} [M]^p) \quad (5.37)$$

$$[B] = \frac{1}{p} z_c^{-1} [M] [A(z_c)]^2 ([U] - z_c^{-p} [M]^p) - z_c^{-p} [M]^p [A(z_c)] \quad (5.38)$$

Naturalmente tem-se ainda as matrizes gigantescas para guardar. A envoltória completa sem pulo de tempo é encontrada fazendo a razão $p = 1$. Portanto, deixa-se esta abordagem e adota-se uma solução mais compacta. Por enquanto se esquece o problema do pulo e concentra-se em aproximar $A(z_0)$ em torno de z_c . Desta forma só é necessário guardar matrizes menores. Para tanto, considera-se a aproximação da equação algébrica em torno de z_c :

$$a = \frac{1}{1 - zz_cm} = \frac{1}{1 - z_cm} + \frac{z_cm(z-1)}{1 - z_cm} \frac{1}{1 - z_cm} + \left(\frac{z_cm(z-1)}{1 - z_cm} \right)^2 \frac{1}{1 - z_cm} + \dots \quad (5.39)$$

Utilizando este resultado na seguinte equação:

$$[V]^r = [T_i] ([U] - z^{-1} [M])^{-1} [T_r] [V]^i \quad (5.40)$$

Assim de modo análogo:

$$[V]^r = [T_i] [\Psi] [T_r] [V]^i \quad (5.41)$$

onde $[\Psi] = [A(z_c)] + z_c^{-1} (z^{-1} - 1) [M] [A(z_c)]^{-2} + \dots$

Agora pode-se fazer uma aproximação da ordem que se deseja. Como exemplo, utiliza-se nessa seção uma aproximação até a terceira ordem.

A aproximação de ordem zero é:

$$[V]^r = [T_i] \{ [U] - z_c^{-1} [M] \}^{-1} [T_r] [V]^i = [T_i] [A(z_c)] [T_r] [V]^i \quad (5.42)$$

Isto resulta na equação:

$$[V]^r = [A] [V]^i \quad (5.43)$$

A aproximação de primeira ordem é:

$$[V]^r = [T_i] [A(z_c)] [T_r] [V]^i + z_c^{-1} (z^{-1} - 1) [T_i] [M] [A(z_c)]^2 [T_r] [V]^i \quad (5.44)$$

Isto resulta na equação:

$$[V]^r = [A] [V]^i + z^{-1} [B] [V]^i \quad (5.45)$$

Onde:

$$[A] = [T_i] \{ [U] - z_c^{-1} [M] [A(z_c)] \} [A(z_c)] [T_r] \quad (5.46)$$

$$[B] = z_c^{-1} [T_i] [M] [A(z_c)]^2 [T_r] [V]^i \quad (5.47)$$

$$\begin{aligned} [V]^r &= [T_i] [A(z_c)] [T_r] [V]^i + z_c^{-1} (z^{-1} - 1) [T_i] [M] [A(z_c)]^2 [T_r] [V]^i + \\ &+ z_c^{-2} (z^{-1} - 1)^2 [T_i] [M]^2 [A(z_c)]^3 [T_r] [V]^i \end{aligned} \quad (5.48)$$

Isto resulta na equação da envoltória compacta:

$$[V]^r = [A] [V]^i + z^{-1} [B] [V]^i + z^{-2} [C] [V]^i \quad (5.49)$$

Em que:

$$[A] = [T_i] [\Omega_1] [T_r] [V]^i$$

$$[B] = z_c^{-1} [T_i] [\Omega_2] [T_r] [V]^i$$

$$[C] = z_c^{-2} [T_i] [\Omega_3] [T_r] [V]^i$$

Onde

$$[\Omega_1] = [U] - z_c^{-1} [M] [A(z_c)] + z_c^{-2} [M]^2 [A(z_c)]^2$$

$$[\Omega_2] = [M] [A(z_c)]^2 - 2z_c^{-2} [M]^2 [A(z_c)]^3 \text{ e}$$

$$[\Omega_3] = [M]^2 [A(z_c)]^3$$

A envoltória complexa, assim como as soluções para aceleração aqui apresentadas não resolvem o problema de simulações em grande dimensões, visto que o armazenamento matricial ainda continua extremamente grande. Contudo, o encadeamento das técnicas de aceleração com uma técnica de segmentação pode resolver de forma bem eficiente a questão de simulações com discretizações mais refinadas. No próximo Capítulo, mostra-se uma técnica de segmentação, a qual consiste na “quebra” do problema global em problemas locais.

6– MODELAGEM TLM-JSN UTILIZANDO TÉCNICAS DE SEGMENTAÇÃO

6.1– INTRODUÇÃO

O método de diakoptica (*do grego, dia = sistema + koptic = cortar*) foi originalmente criado em meados da década de 60 por Kron [25, 62]. A técnica de diakoptica ou segmentação consiste em um método para resolução de problemas físicos de grande escala. Ou seja, o particionamento de uma larga estrutura em várias sub-estruturas, as quais são resolvidas independentemente e reconectadas posteriormente. A diakoptica em redes é bem conhecida na solução de equações de estado, elementos finitos e análises de estruturas [63–66]. A técnica torna-se mais atrativa quando as estruturas possuem muitas partes repetidas e apenas algumas diferenças. Esta técnica tem sido extensivamente aplicada em conjunção com métodos no domínio da frequência [67–69]. E, posteriormente, foi apresentado por P. B. Johns e Akhtarzad em 1981 uma extensão para o método TLM no domínio do tempo [70, 71]. P. B. Johns e Akhtarzad mostraram como sub-estruturas podem ser resolvidas no domínio do tempo usando o método TLM e como reconectá-las posteriormente. Neste capítulo, apresenta-se a técnica de segmentação na modelagem TLM-JSN e seus aspectos mais básicos.

6.2– DIAKOPTICAS NA MODELAGEM TLM-JSN

Ao discretizar uma estrutura em vários segmentos pode-se acelerar sua resolução com menor esforço computacional. A Fig. 6.1 mostra uma malha TLM e sua subdivisão em blocos. Cada subdivisão é considerada como uma simulação adicional e uma fonte de entrada para os outros blocos. A resolução do sistema que compõe o problema deverá ser

efetivamente processada em paralelo e com isso diminuindo o esforço computacional com sistemas em menores dimensões.

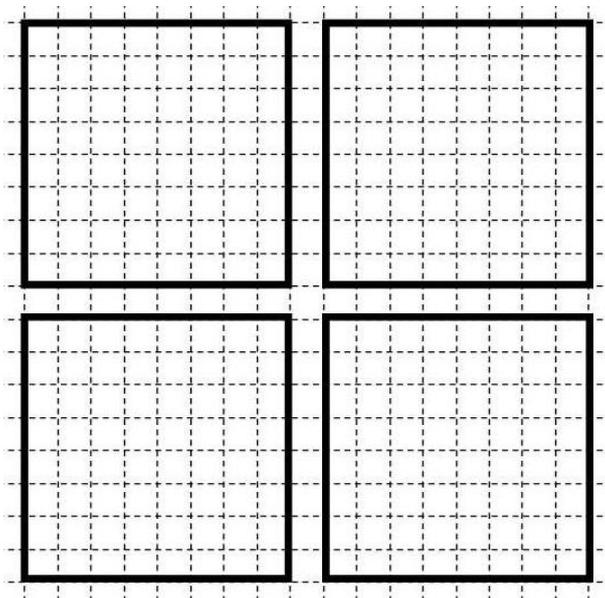


Figura 6.1: Representação em blocos de técnica de segmentação no método TLM bidimensional.

As portas que ficaram à margem dos blocos são utilizadas para repassar informações entre o resto da malha, ou seja, tornando-se fonte de excitação para os demais blocos. Cada sub-estrutura possui suas peculiaridades, e as matrizes que modelam cada sub-estrutura apresentam a característica da estrutura analisada em cada bloco. Como todos os outros métodos existem desvantagens, a implementação generalizada para qualquer estrutura é complexa devido a interligação dos blocos e a criação de suas respectivas matrizes.

A solução no domínio do tempo de redes pelo método TLM é completada repassando-a energia armazenada nos componentes da rede pelas linhas de transmissão. Isso introduz um atraso de tempo nos pulsos que viajam pelas linhas. Logo, é conveniente analisar as estruturas em termos de pulsos incidentes V^i e refletidos V^r , sobre as linhas de transmissão nos nós da rede. Assim, para um k -ésimo intervalo de tempo,

$$[V]_k^r = [S][V]_k^i + [V]_k^s \quad (6.1)$$

$$[V]_{k+1}^i = [C][V]_k^r \quad (6.2)$$

onde $[S]$ é a matriz de espalhamento para todos os nós da rede, $[C]$ é a matriz de conexão que descreve a topologia da rede, e $[V]^s$ é um vetor fonte.

O método de diakópticas é aplicado na modelagem TLM-JSN partindo das equações (4.3) e (4.4) que são descritas no Capítulo 4.

$$[in V]_k^i = [T^t][ex V]_k^i + [C][S][in V]_{k-1}^i \quad (6.3)$$

$$[ex V]_k^r = [T][S][in V]_k^i \quad (6.4)$$

As equações (6.3) e (6.4) representam fisicamente a mesma característica da expressão utilizada em [70, 71]. As tensões internas sofrem influência do comportamento interno da malha ($[C][S][in V]_{k-1}^i$), porém adicionando as fontes externas ($[T^t][ex V]_k^i$). Portanto, a aplicação do método de segmentação no TLM-JSN é viável, visto que a interpolação linear contabiliza tais efeitos. No entanto a utilização de interpolação linear para controlar quantos passos de tempo serão omitidos (p) trás um problema na utilização de diakópticas. A cada salto de tempo perde-se um percentual em energia do pulso. Este problema ainda encontra-se em estudo. Vale ressaltar que para $p = 1$ o método TLM-JSN torna-se idêntico ao TLM convencional e a técnica funciona perfeitamente.

Contudo, foi identificado uma possível razão para essa perda. As tensões incidente e refletidas omitidas na interpolação não são contabilizadas em cada iteração e, devido a omissão dessas tensões, perde-se um fator de energia relacionado com a razão p e o número de sub-estruturas. Para solucionar esse problema pode-se adotar uma matriz que relaciona as tensões refletidas com a interconexão global entre os blocos. Essa matriz $[\Xi]$ que relaciona as tensões globais das sub-estruturas com as tensões refletidas torna-se um termo adicional na equação (6.3). Logo, o sistema torna-se:

$$[in V]_k^i = [T^t][ex V]_k^i + [\Xi][ex V]_{k-1}^r + [C][S][in V]_{k-1}^i \quad (6.5)$$

Aplicando a transformada Z e fazendo algumas operações algébricas em (6.5) e substituindo em (6.4) chega-se a

$$[ex V]_k^r = (I - [\Omega])^{-1} [\Psi][ex V]_k^i \quad (6.6)$$

onde $[\Omega] = [T][S](I - z^{-1}[C][S])^{-1}[T^t]$ e $[\Psi] = z^{-1}[T][S](I - z^{-1}[C][S])^{-1}[\Xi]$. Nota-se que a expressão trata o domínio externo recebendo informações do domínio interno através das matrizes de espalhamento $[S]$ e topologia $[C]$. A Fig. 6.2 mostra o comportamento deste

feito para razões de salto (p) diferentes em uma sub-estrutura com n portas. Nota-se que as tensões incidente e refletidas são omitidas para razões $p > 2$, e com isso aumentado a perda nas $k\Delta t$ iterações.

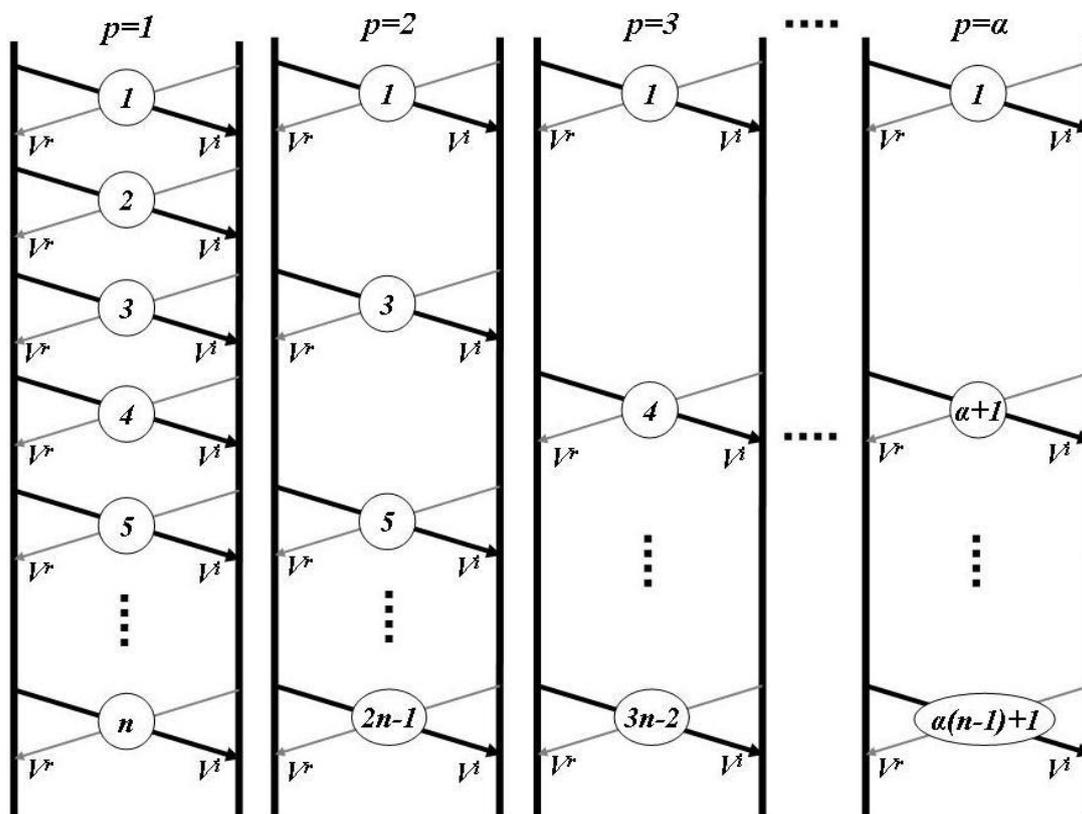


Figura 6.2: Representação das perdas de energia na aplicação da técnica de diakoptica no método TLM-JSN

A Fig. 6.3 mostra o fluxograma do Método TLM-JSN usando a técnica de diakoptica. É muito similar ao fluxograma do TLM, com a diferença nas leituras de entrada, divisão da estrutura em sub-estruturas e resolução do sistema. Cada sub-estrutura possui matrizes específicas, porém a resolução do sistema é feita de forma simultânea, tornando menos desperdicioso o processamento.

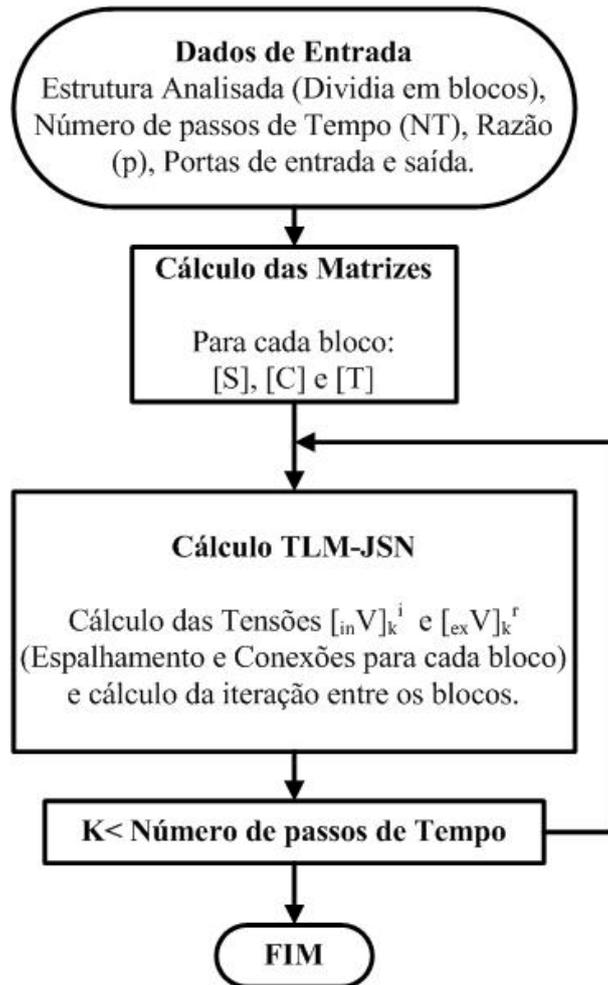


Figura 6.3: Fluxograma do algoritmo TLM-JSN.

7– SIMULAÇÕES E RESULTADOS

7.1– INTRODUÇÃO

Para validar a teoria desenvolvida foram realizadas simulações e estas comparadas com resultados pré-existentes. O pseudo-código implementado em MATLAB foi testado em diversas estruturas. Foi adotado como parâmetro de referência o pulso gaussiano. Esse pulso foi injetado em diversas estruturas, com características de filtros de microondas. No entanto, a análise consiste em comparar a resposta das estruturas a esse pulso gaussiano usando as técnicas apresentadas anteriormente. A Fig. 7.1 mostra a amplitude do pulso gaussiano construído ao longo de 2000 passos de tempo e centralizado em 1000. A magnitude também é mostrada na Fig. 7.1, os passos de tempo nesse caso são normalizados em frequência de acordo com o maior passo de tempo.

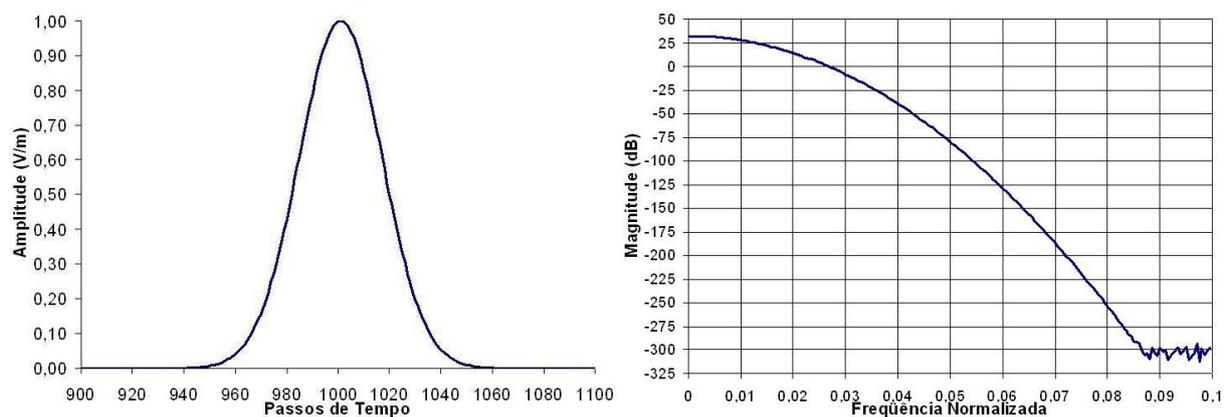


Figura 7.1: *Amplitude do pulso gaussiano e sua Magnitude*

Na seção seguinte apresentam-se as estruturas e os resultados obtidos.

7.2– ESTRUTURAS ANALISADAS

7.2.1– Filtro WR28

O guia de onda WR28 opera na banda R de frequência que vai de 26,5 – 40GHz. A Fig. 7.2 mostra a primeira estrutura discretizada por 10×50 células TLM. Foi considerado um meio sem perdas, com permissividade e permeabilidade do vácuo e as dimensões $\Delta l = \Delta x = \Delta y = 1$ un. Paredes magnéticas são utilizadas para evitar efeitos semelhantes àqueles existentes nas bordas dos capacitores de placas paralelas, quando submetidos a um campo eletromagnético. Considerou-se uma excitação gaussiana em forma de pulso em uma porta da malha TLM.

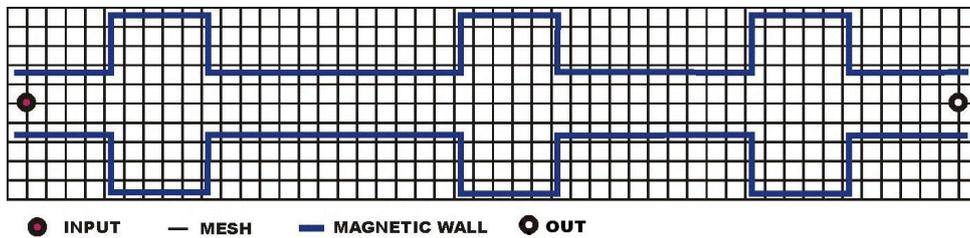


Figura 7.2: Filtro utilizado para a simulação TLM

Os resultados obtidos, mostrados na Fig. 7.3, são comparados com o Software Mefisto-2D desenvolvido por Hofer [52]. O TLM-JSN utilizado nessa simulação não fez uso da redução de modelo, mas somente do algoritmo de saltos de tempo controlados.

Percebe-se, pela figura, que para uma razão p igual a 2 passos de tempo o TLM-JSN comporta-se de forma similar ao TLM convencional. Os erros estão abaixo de 0,001%.

O comportamento do TLM-JSN ao variar a razão p é exibido na Fig. 7.4. Para tornar as simulações mais velozes utilizou-se a redução de modelo ROM-1. Nota-se que a interpolação linear empregada segue de forma adequada o comportamento da amplitude independente do valor de p . Quando a razão de saltos de tempo é igual a 30 observa-se que há algumas diferenças nos valores de amplitude quando comparadas ao método convencional. Isso se deve à largura de banda do pulso gaussiano. Quanto mais larga, melhor será a aplicação do TLM-JSN.

O resultado da utilização do TLM-JSN com ROM-2 é mostrado na Fig. 7.5. Nota-se

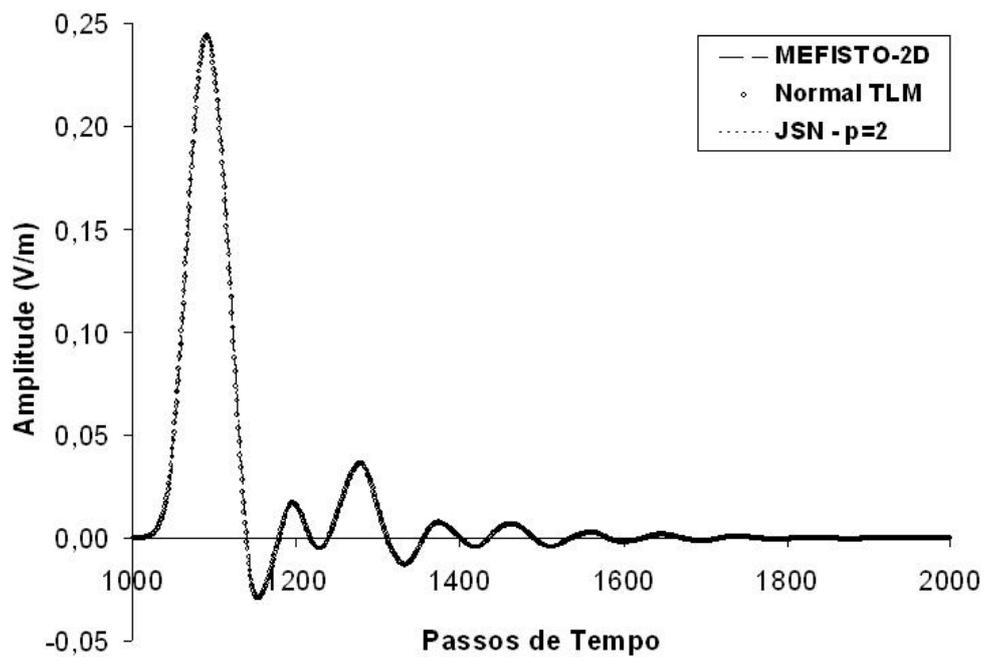


Figura 7.3: Comparação entre as TLM Normal, TLM (MEFISTO) e TLM-JSN ($p = 2$), para a estrutura da Fig. 7.2

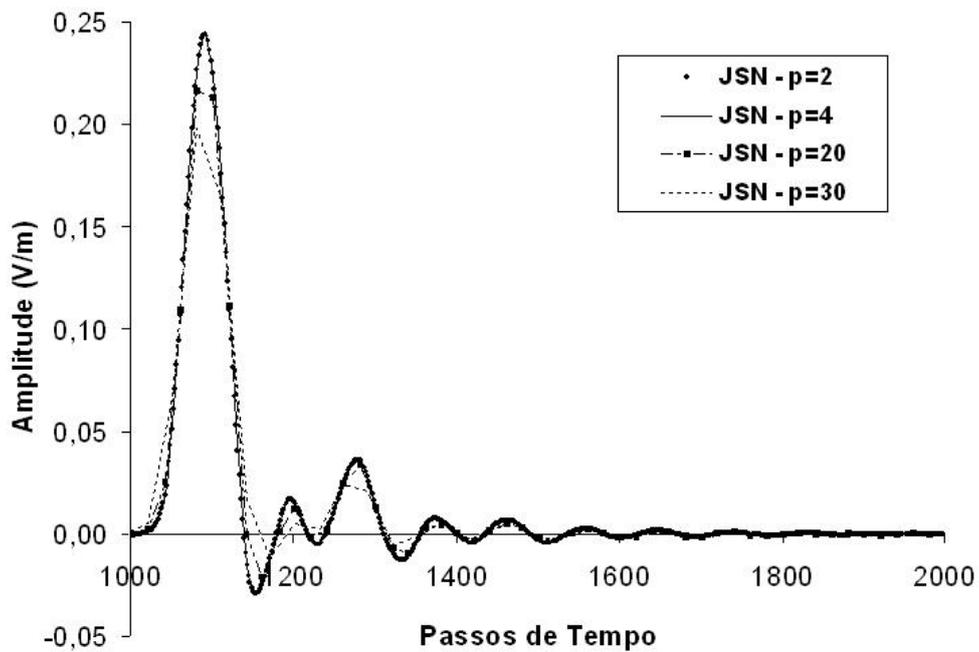


Figura 7.4: Comparação do TLM-JSN para diversas razões p para a estrutura da Fig. 7.2

que para p saltos de dez passos de tempo o comportamento desta versão do método é muito próximo ao do método convencional. Alterando-se o valor de p e controlando o número de autovalores de acordo com a banda desejada pode se obter resultados precisos e consideravelmente rápidos.

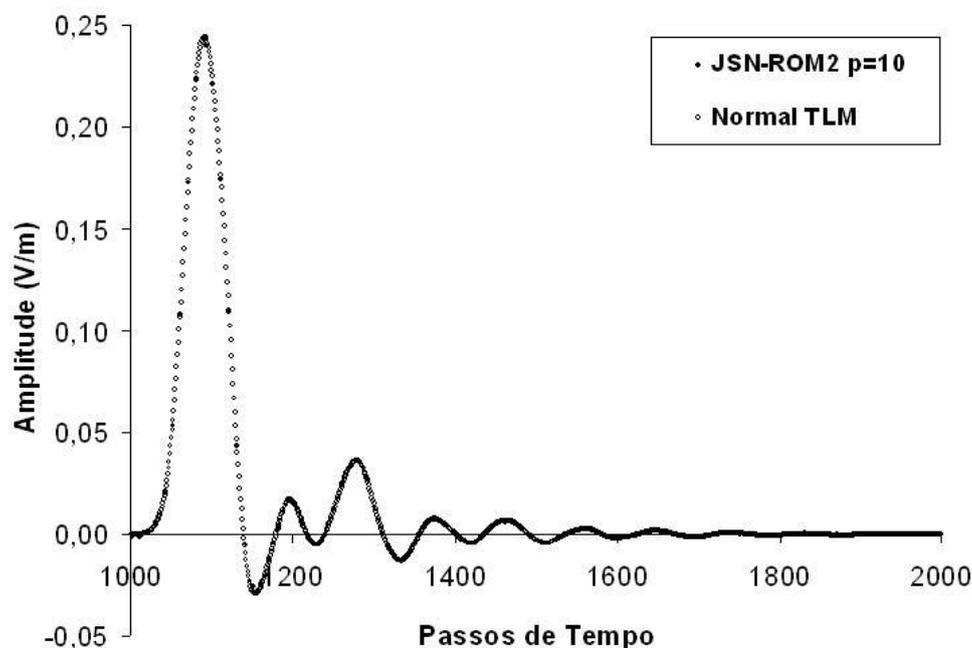


Figura 7.5: Comparação do TLM-JSN com ROM-2 e o TLM Normal para a estrutura da Fig. 7.2

Uma outra análise constituiu no estudo do tempo de processamento entre as versões do TLM-JSN e o TLM usual, ao variar a razão p . A Fig. 7.6 apresenta os resultados. As versões do TLM-JSN são referidas na figura como: algoritmo de saltos de tempo controlados sem redução de modelo (TLM-JSN), ROM-1 e ROM-2.

Por meio dessa figura é que verifica-se o desempenho superior do segundo algoritmo de redução de modelo (ROM-2). A diferença no tempo médio de simulação entre o algoritmo TLM usual e o ROM-2 é de aproximadamente 1900% em média. A inovação desta técnica é o modo de controle dos autovalores para obter-se a precisão requerida. O TLM-JSN sem redução de modelo apresenta ganhos de processamento para $p > 150$.

Através do uso da Transformada Rápida de Fourier (FFT), na resposta da estrutura analisada no domínio do tempo, é que se observa a necessidade de que a mesma tenha que ser passa-baixas. A Fig. 7.7 exemplifica melhor isso.

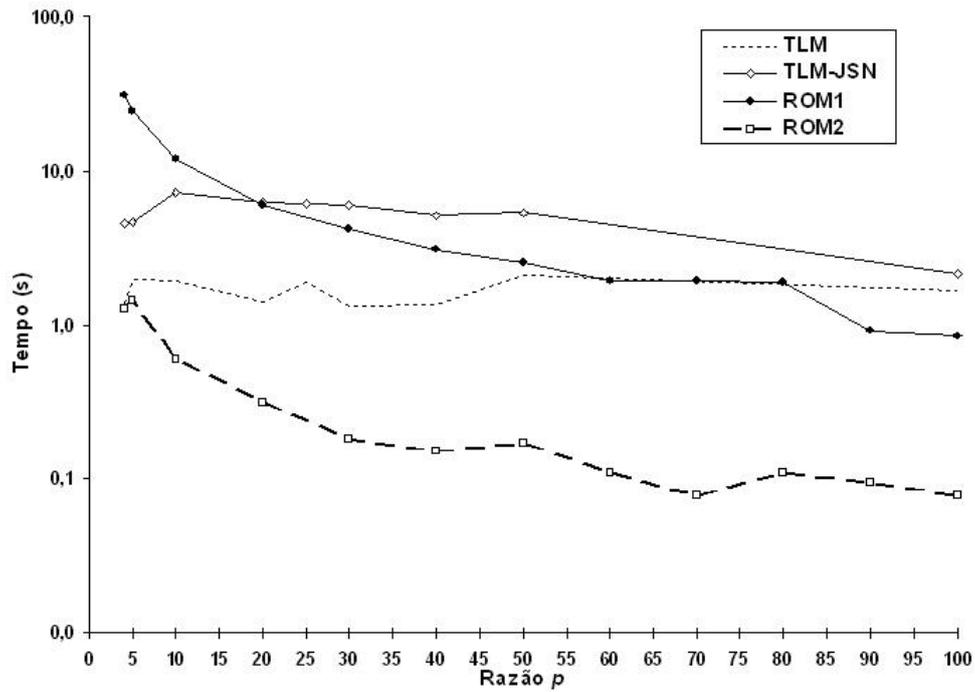


Figura 7.6: Comparação dos tempos de processamento entre as versões do TLM-JSN e do TLM normal para a estrutura da Fig. 7.2

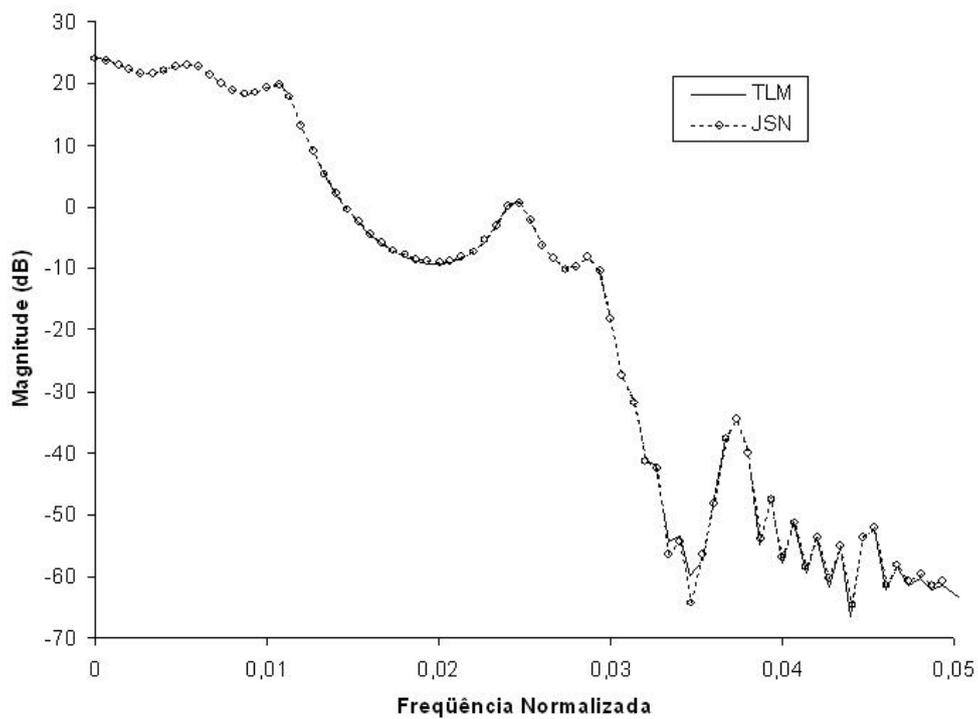


Figura 7.7: Espectro de Frequências da resposta obtida para a estrutura da Fig. 7.2, ($p = 2$)

7.2.2– Filtro IRIS Tipo A

A segunda estrutura analisada é exibida na Fig. 7.8. Na literatura é conhecida como Íris. Foi utilizada as mesmas características de simulação da seção anterior, onde foi usada um meio sem perdas, com permissividade e permeabilidade do vácuo e as dimensões $\Delta l = \Delta x = \Delta y = 1$ un.

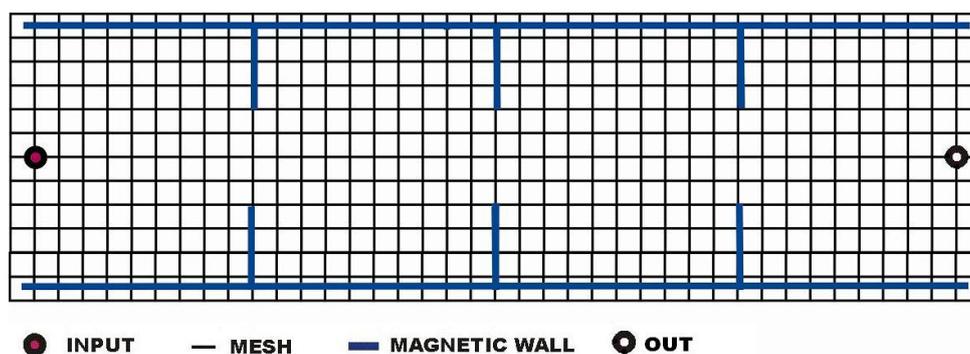


Figura 7.8: Filtro Íris utilizado para a simulação TLM.

A discretização TLM da Íris gerou uma malha de 10 por 40 nós. A entrada da excitação também é mostrada na Fig. 7.9. Utilizou-se novamente um pulso Gaussiano.

A primeira análise da estrutura consistiu em comparar os resultados obtidos através da aplicação do TLM normal e do TLM-JSN, por meio das versões sem e com redução de modelo com $p = 16$ (Fig. 7.9).

Pela figura nota-se que, a menos de certos passos de tempo, os algoritmos do TLM-JSN acompanham perfeitamente o resultado do TLM convencional. Aqui, mais uma vez, comprova-se a eficácia do método proposto.

Também foram efetuados mais dois tipos de análise. A primeira consistiu em verificar o comportamento do TLM-JSN sem redução de modelo, ao variar a razão p . A segunda comparou os tempos de processamento entre as versões do JSN e do TLM usual. As Fig. 7.10 e Fig. 7.11 exibem os resultados da primeira e da segunda análise, respectivamente.

Pela Fig. 7.12, vê-se que para p igual a trinta, o TLM-JSN diverge do TLM normal em mais passos de tempo do que para as outras razões p . Já é por meio da Fig. 7.11 que se evidencia mais uma vez a superioridade do TLM-JSN-ROM-2 sobre os demais algoritmos. A diferença na média de tempo entre o ROM-1 e o ROM-2 é de 1700%. Além

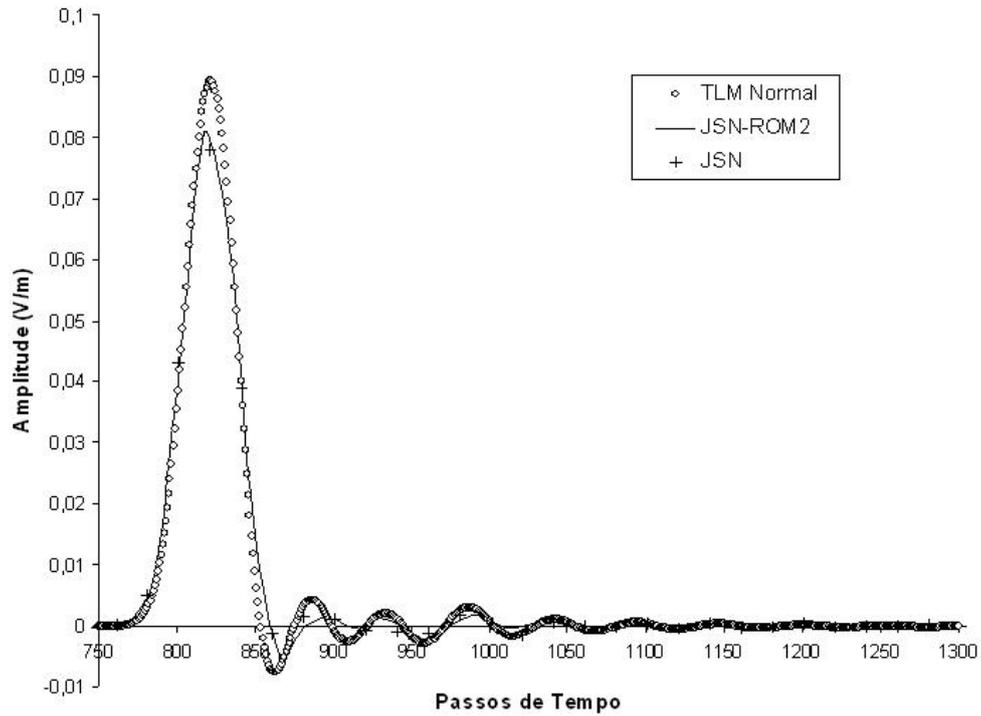


Figura 7.9: Comparação entre o TLM Normal, TLM -JSN ($p = 16$) e TLM-JSN-ROM-2 ($p = 16$), para a estrutura da Fig. 7.8

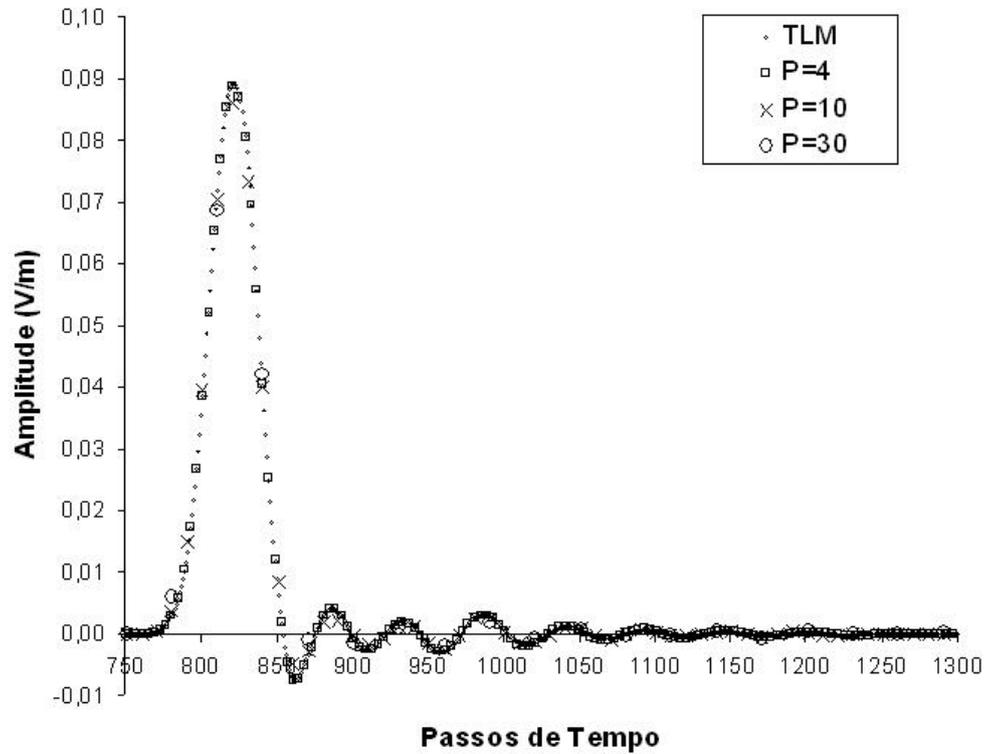


Figura 7.10: Comparação do TLM-JSN sem ROM para diversas razões p , para a estrutura da Fig. 7.8

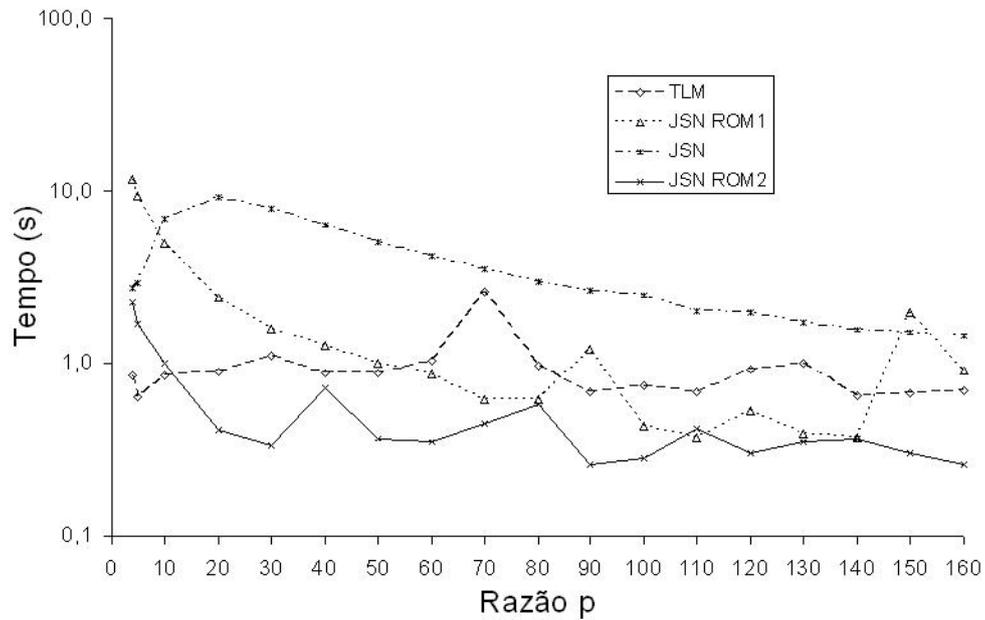


Figura 7.11: Comparação dos tempos de processamento entre as versões do TLM-JSN e do TLM normal, para a estrutura da Fig.7.8

de proporcionar resultados precisos, os mesmos são obtidos com maior velocidade. A Fig. 7.12 apresenta o espectro de frequência da resposta obtida para o Filtro Íris.

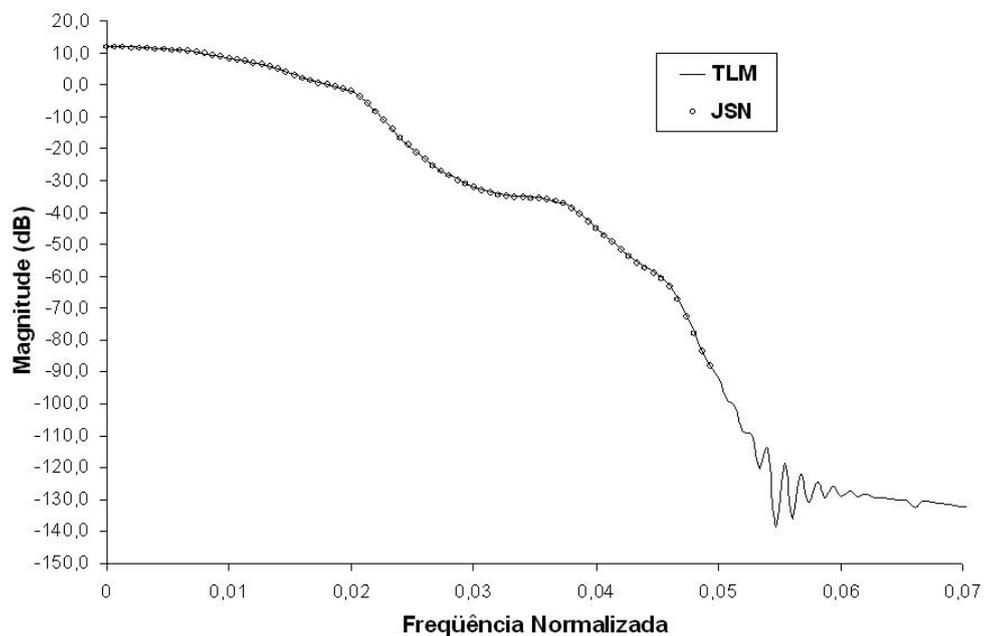


Figura 7.12: Espectro de Frequências da resposta obtida para a estrutura da Fig. 7.8, ($p = 30$)

O número de pontos reduzidos do JSN nessa figura se deve ao menor número de passos de tempo utilizados na simulação.

7.2.3– Filtro IRIS Tipo B

A última estrutura analisada é mostrada na Fig. 7.13. Para validar o método, esta também é uma estrutura do tipo Iris. Ela é ainda uma variação do Filtro Íris da Fig. 7.8. Foi considerado um meio sem perdas, com permissividade e permeabilidade do vácuo e as dimensões $\Delta l = \Delta x = \Delta y = 1$ un.

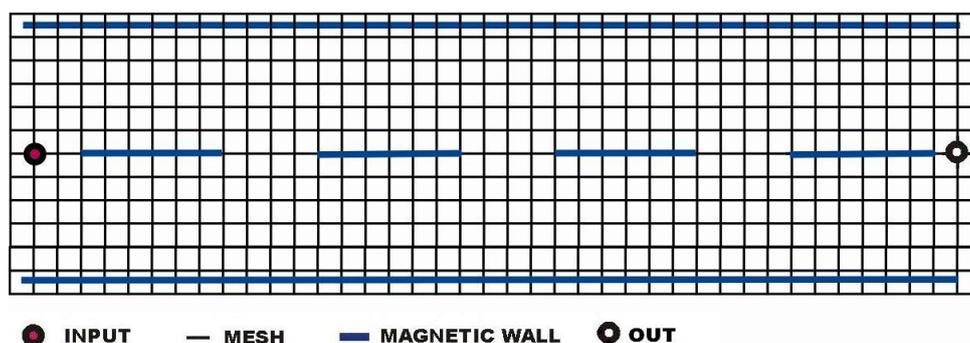


Figura 7.13: *Tipo de Filtro Íris utilizado para a simulação TLM.*

A discretização TLM dessa Íris gerou novamente uma malha de 10 por 40 nós. A entrada é mostrada e a excitação é outro pulso Gaussiano.

Só dois tipos de estudo foram realizados nesta estrutura. No primeiro, variou-se a razão p para conferir o comportamento do TLM-JSN-ROM-1 (Fig. 7.14). Já no segundo compararam-se os tempos de processamento entre as versões com redução de modelo do TLM-JSN e do TLM convencional (Fig.7.15).

As duas figuras anteriores comprovam mais uma vez que o método desenvolvido é válido e destaca ainda a velocidade da segunda redução de modelo. Para efeito de comparação a diferença entre os tempos médios de processamento entre o ROM-1 e o ROM-2 é cerca de 1300%.

Só para verificar novamente a banda passante da resposta obtida, obteve-se o espectro da mesma, como exibido na Fig. 7.16. Também foram utilizados números reduzidos de passos de tempo nessa simulação.

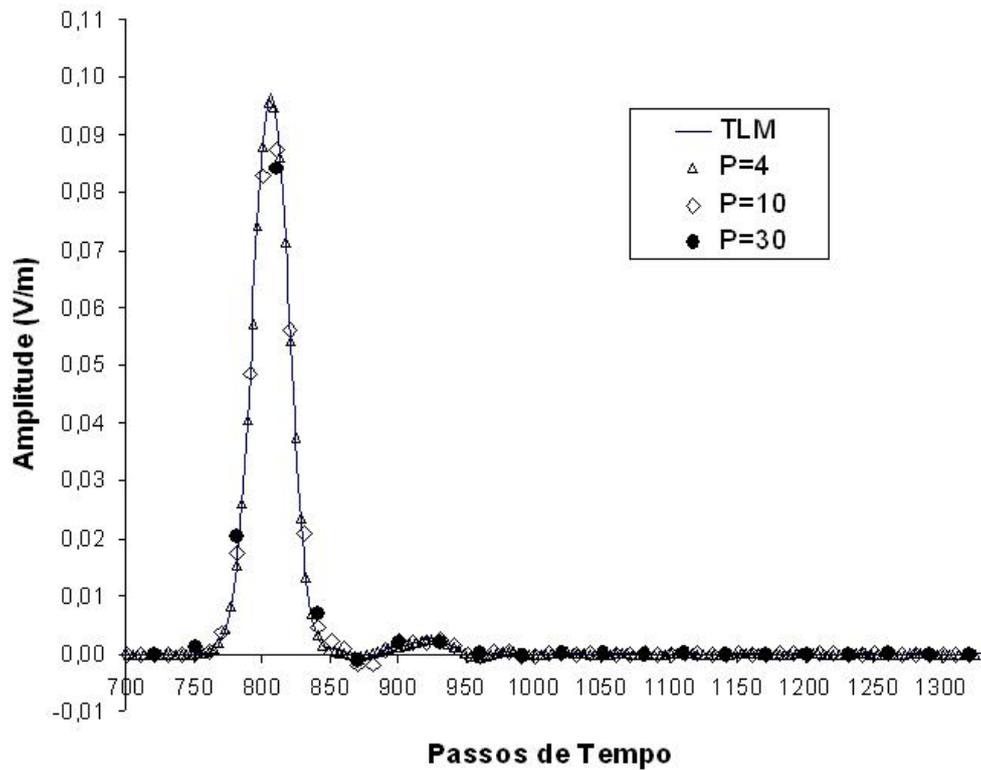


Figura 7.14: Comparação do TLM-JSN-ROM-1 para diversas razões p , para a estrutura da Fig. 7.13

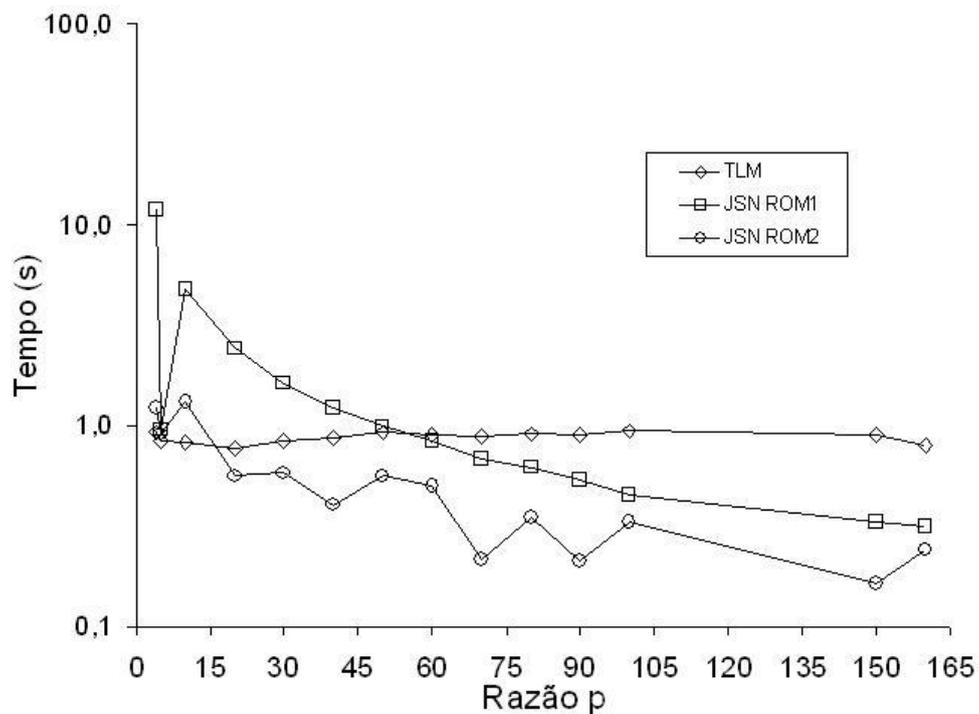


Figura 7.15: Comparação dos tempos de processamento entre as versões do TLM-JSN com ROM e do TLM normal, para a estrutura da Fig. 7.13

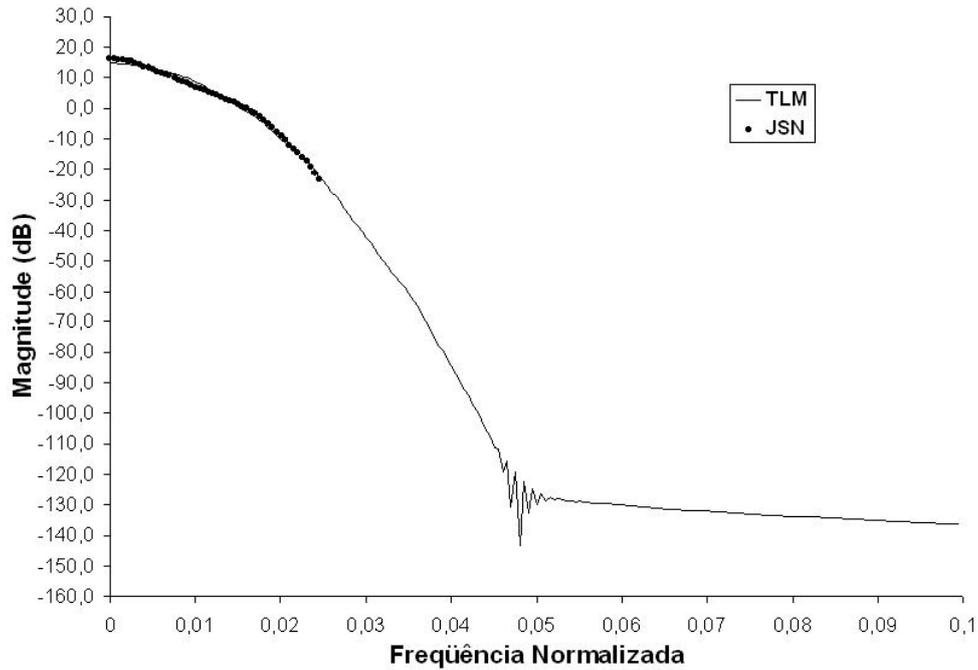


Figura 7.16: Espectro de Freqüências da resposta obtida para a estrutura da Fig. 7.13, ($p = 10$)

7.2.4– Envoltória Complexa

Na simulação da envoltória complexa aplicada no TLM-JSN, foi utilizado o guia de onda tipo WR28, discretizado com 10×50 células, onde foi considerado um meio sem perdas, com permissividade e permeabilidade do vácuo e as dimensões $\Delta l = \Delta x = \Delta y = 1$ un..

Na simulação, o ponto de excitação é o mesmo adotado na 7.2.1. E também foi simulado um ambiente vazio discretizado com 10×20 células com as duas técnicas apresentadas.

As duas técnicas mostraram-se muito precisas como pode-se notar nas Figs. 7.17 e 7.18, visto que as técnicas são matricialmente diferentes em relação ao tamanho de sua composição e seus aspectos de resposta ao pulso gaussiano foram bem precisos.

Futuramente, com a implementação de diakóptas na modelagem de envoltória complexa pode-se fazer simulações de problemas mais complexos devido ao menor armazenamento matricial e ainda trazendo o benefício de simular na freqüência desejada.

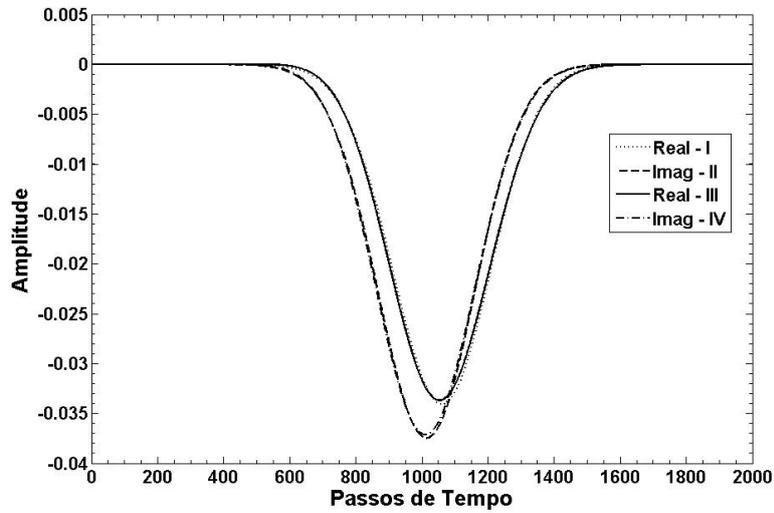


Figura 7.17: Resposta em amplitude para o guia de onda WR28 com a técnica de envelope complexo. As curvas I e II foram obtidas com envelope completo e as curvas III e IV com o compacto.

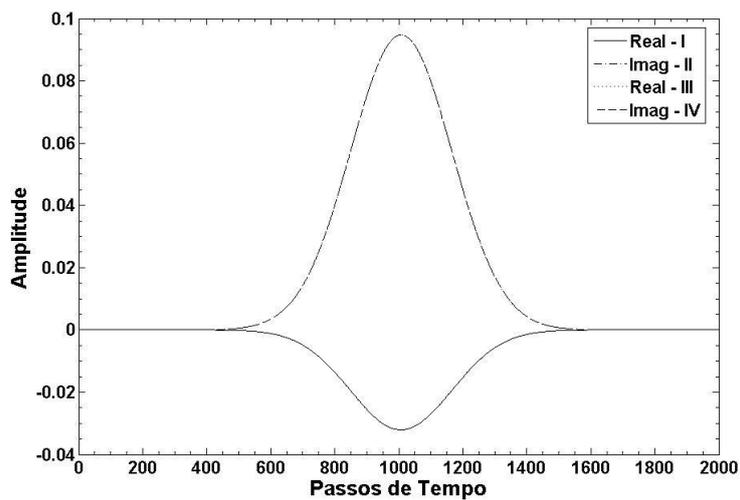


Figura 7.18: Resposta em amplitude para o espaço vazio com a técnica de envelope complexo. As curvas I e II foram obtidas com envelope completo e as curvas III e IV com o compacto.

7.2.5– Diakópticas

As simulações usando a técnica de segmentação são apresentadas nessa seção. Foi utilizado o TLM e TLM-JSN com perdas. A Fig. 7.19 mostra a estrutura e sua respectiva discretização. Foram utilizadas estruturas com paredes elétricas, coeficientes de permissividade do vácuo e condutividade do cobre ($5,76 \cdot 10^{-7}$ S/m). O número de passos de tempo ($k = 2000$) e $\Delta x = \Delta y = 0,1 \mu\text{m}$.

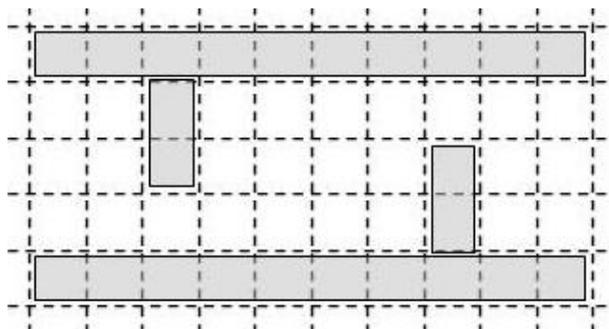


Figura 7.19: Estruturas simuladas com TLM e TLM-JSN com perdas.

A Fig. 7.20 mostra o resultado para o método TLM e TLM-JSN com perdas. Entretanto não foi aplicado a técnica de segmentação. Percebe-se que o TLM e TLM-JSN possuem uma excelente aproximação para uma razão $p = 2$, $p = 4$, $p = 5$ e $p = 10$.

A Fig. 7.21 mostra a resposta em amplitude da mesma estrutura simulada anteriormente usando a técnica de segmentação. Foi utilizado os mesmos parâmetros de simulação, porém a estrutura foi dividida em dois segmentos. Nota-se que a imprecisão do método TLM-JSN é diretamente proporcional a razão p . Contudo, a energia perdida usando a técnica de diakópticas está ligada diretamente com as tensões internas incidentes nas conexões de fronteira entre os blocos.

A Fig. 7.23 mostra a resposta em amplitude da mesma estrutura simulada. Foi utilizado os mesmos parâmetros de simulação, porém a estrutura foi dividida em 4 segmentos e aumentada sua discretização para 10×10 células. Percebe-se claramente que o padrão de resposta decaí mais rapidamente do que com dois blocos, a conexão entre quatro blocos acarreta em perdas maiores. Logo, o aumento consecutivo dos blocos e o aumento das razões de saltos então diretamente ligados a perda dessa energia como foi proposto no Capítulo 6. As simulações foram todas realizadas utilizando o programa MATLAB em

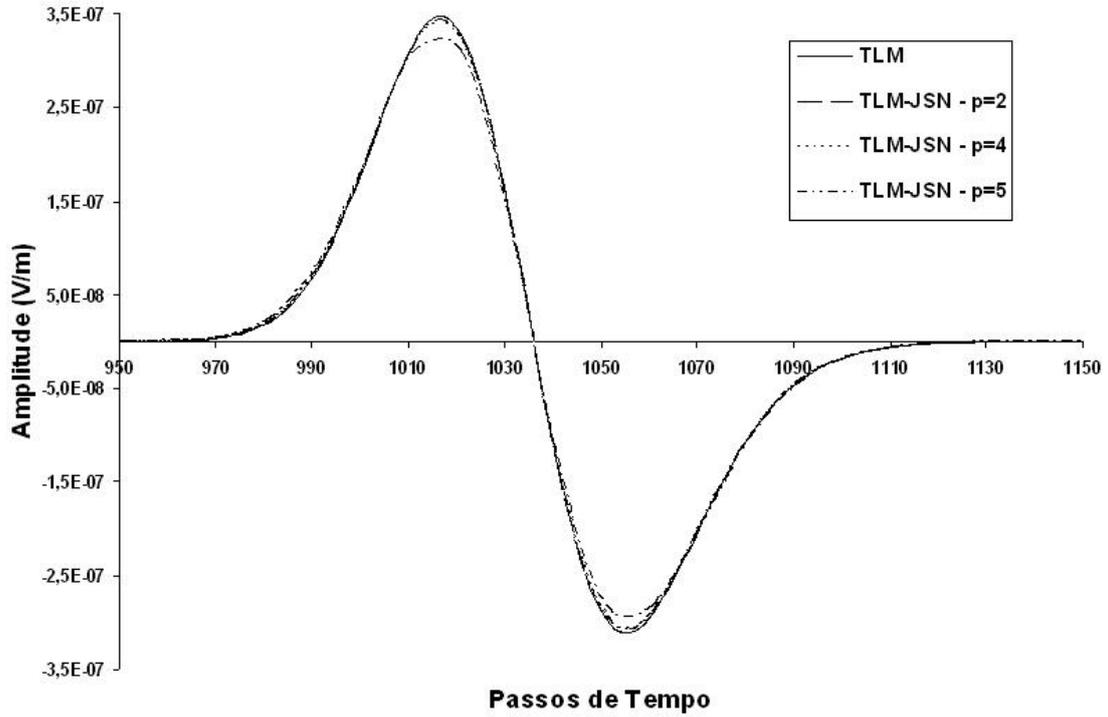


Figura 7.20: Resposta em Amplitude para a estrutura da Fig. 7.19, usando TLM e TLM-JSN com perdas com diferentes razões de salto.

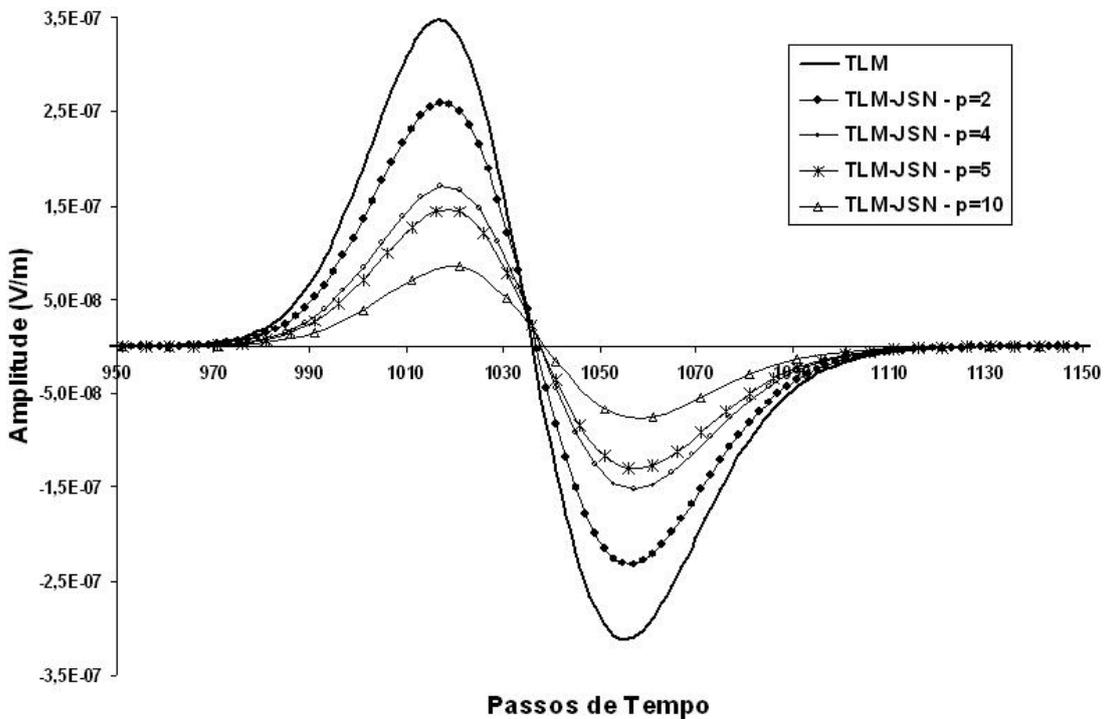


Figura 7.21: Resposta em Amplitude usando TLM e TLM-JSN com perdas e técnica de dia-kópticas. Os blocos possuem 5×5 células cada.

computador AMD-Atlhon 2.4GHz, com 512MB de RAM.

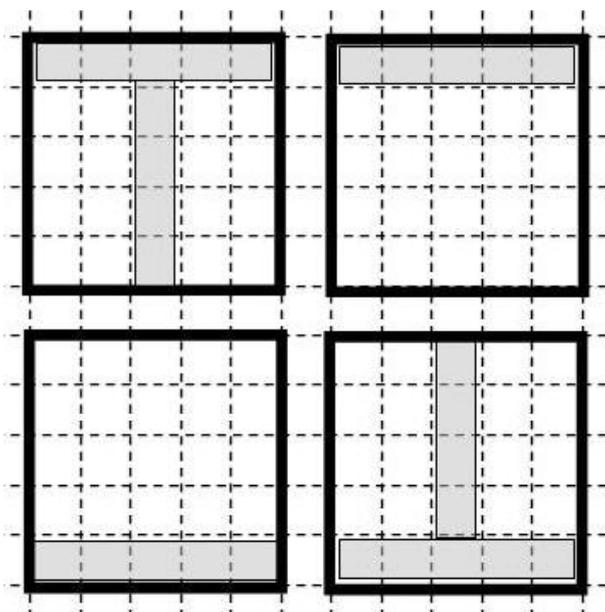


Figura 7.22: Estrutura simulada com TLM e TLM-JSN com perdas e técnica de diakopticas usando quatro blocos, possuindo 5×5 células cada com $\Delta x = \Delta y = 0,05\mu m$.

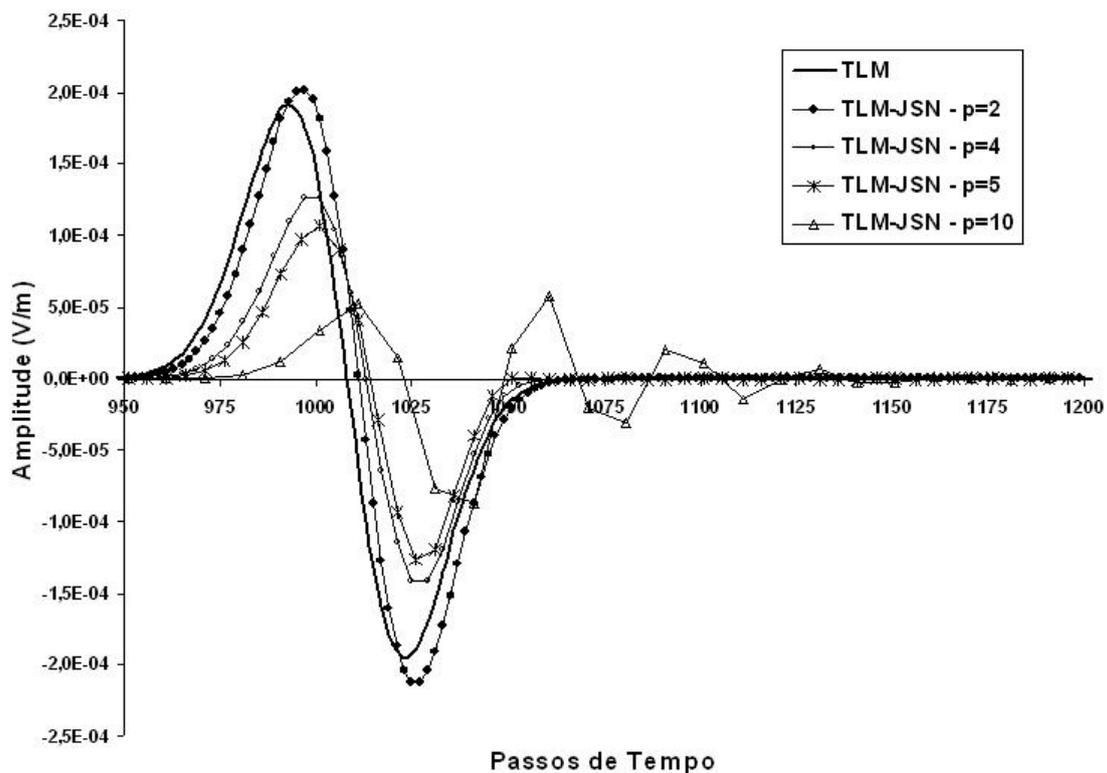


Figura 7.23: Resposta em Amplitude usando TLM e TLM-JSN com perdas e técnica de diakopticas usando quatro blocos.

8– CONCLUSÃO

O método numérico TLM atua no domínio do tempo e pode analisar diversos tipos de estruturas eletromagnéticas. Este trabalho apresentou uma nova versão do método TLM com o intuito de realizar simulações com maior velocidade. Todo o procedimento empregado teve por base o TLM-2D convencional. Três estruturas foram analisadas com a técnica desenvolvida. Os resultados obtidos comprovam a validade do método e sua vantagem sobre o método convencional. Além disso, esses resultados reafirmam que uma das aplicações principais do TLM-JSN é na utilização com ferramentas CAD (Computed Aided Design) para simulação de projetos de microeletrônica. Antes da implementação deste método era necessário um grande volume de armazenamento e tempo computacional para análise eletromagnética.

Como observado a eficiência do TLM-JSN depende fundamentalmente da largura de banda da excitação. Quanto mais larga for essa largura de banda, poderá se utilizar maiores razões p na interpolação linear de primeira ordem. Com isso, a diferença de tempo na simulação será mais significativa. Uma ferramenta computacional para o auxílio nas realizações de simulações foi criada. Ela tornará a análise de uma estrutura possível, não levando em consideração o processador utilizado pelo usuário.

O trabalho concluído servirá como base para novas pesquisas que vêm sendo conduzidas, tendo como fundamento análises mais complexas com o TLM-JSN, bem como a expansão para o caso tridimensional e análise de estruturas com comportamento não-linear. Além da análise de circuitos integrados, a nova técnica, quando aprimorada, pode ser empregada ainda no estudo de propagação de ondas eletromagnéticas de Estações Rádio Base (ERBS), para verificar a cobertura celular de uma dada região em um curto período de tempo.

8.1– PROPOSTA DE DESENVOLVIMENTO FUTURO

A implementação do método TLM-JSN foi expandida no máximo que a modelagem bidimensional permite. Porém, há uma série de implementações computacionais e matemáticas que podem ser melhoradas e sanadas. Por exemplo, a programação do método em uma linguagem C/C++, conjuntamente com um visualizador bidimensional e tridimensional ainda é necessário.

No decorrer da pesquisa foi desenvolvido dois visualizadores de arquivos básicos TLM, um para o caso 2D e outro para 3D, utilizando C++/OPENGL. No entanto, não são suficientes para resolver o problema do visualizador de circuitos, pois a maior dificuldade é estabelecer uma leitura de arquivos do tipo *.CIF* ou *.GDSII*, onde ambos formatos de arquivos são de ferramentas CAD profissionais, como por exemplo o Software Cadence.

Com uma ferramenta de simulação nesse nível é possível implementar um servidor com finalidade de disponibilizar um simulador de circuitos integrados *on-line*. Onde o usuário entra com o projeto do chip em arquivos no formato *.CIF*. Ao executar esta tarefa o visualizador criará imagens do plano superior de cada camada do chip. Então essas imagens geradas serão lidas pelo simulador TLM-JSN e executada a simulação do espalhamento de campo no chip. Criando posteriormente uma imagem com o mapeamento de intensidade de campos e outros parâmetros de análise como susceptibilidade e emissividade em função das frequências de operação.

Outra proposta muito interessante está na resolução do problema de segmentação, que ainda encontra-se em estudo. Uma modelagem mais refinada, levando em consideração as perdas ocorridas nas tensões internas incidentes na interface dos blocos deve ser modelada, visto que o caso tridimensional é mais complexo devido as suas seis faces com $n \times m$ portas em cada face.

A aplicação de diakópticas usando a envoltória complexa pode ser utilizada na simulação de espalhamento de campo eletromagnético em estações rádio-base (ERBs).

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Christopoulos C., "The Transmission-Line Modeling Method TLM", Oxford, IEE PRESS, 1995.
- [2] Christopoulos C., "The application of transmission-line modelling (TLM) to electromagnetic compability problems", IEEE Transaction EMC-35, 185-191, 1993.
- [3] Morente, J. A., J. A. Porti, and M. Khalladi, "Absorbing boundary conditions for the TLM method", IEEE Transaction MTT-40, 2095-2099, 1992.
- [4] Herring, J. L., P. Naylor, and C. Christopoulos, "The application of transmission-line modelling in electromagnetic compability studies", Int. J. Num. Model. 4, 143-152, 1991.
- [5] Naylor, P., C. Christopoulos, and P. B. Johns, "Coupling between electromagnetic fields and wires using TLM", IEE Proc. 134, 679-686, 1987.
- [6] Carvalho Jr. C. A. T., Menezes. Leonardo R. A. X. de, "Extraction of S-Parameter Using the Three-Dimensional Transmission Line Matrix (TLM) Method". SBMO/IEEE MTT-S Int. Micr. and Opt. Conf. - IMOC, Foz do Iguaçu, vol. 2, pp. 967-969, 2003.
- [7] Assunção, L. A., "Amplificador de Potência com Controle Digital em Tecnologia CMOS para aplicação em Telemetria". Dissertação de Mestrado em Eng., pub. 202, Dep. de Engenharia Elétrica, UnB, Brasília, 2004.
- [8] P. B. Johns and R. L. Beurle, "Numerical Solution of 2-Dimensional Scattering Problems Using a Transmission-Line Matrix", Proc. Inst. Elec. Eng., 118(9):1203-1208, September 1971.
- [9] C. Huygens, "Traité de la Lumière", Leiden, 1690, Pierre Vander Aa, reprinted in "Oevres Complètes de Christiaan Huygens", Socièté Hollandaise des Sciences, Vol. 19, Amsterdam 1967, Swets and Zeitlinger.
- [10] J. R. Whinnery, and S. Ramo, "A new Approach to the Solution of High Frequency Field Problems", Proc. I.R.E., Vol. 32, pp. 284-288, May 1944.
- [11] G. Kron, "Equivalent Circuit of the field equations of Maxwell - I", Proc. I.R.E., Vol 32, pp. 289-299, May 1944.
- [12] J. R. Whinnery, C. Concordia, W. Ridgway, and G. Kron, "Network Analyzer Studies of Electromagnetic Cavity Resonators", Proc. I.R.E., Vol. 32, pp. 360-367, June 1944.

- [13] P. B. Johns, "Applications of the Transmission-Line Matrix Method to Homogeneous Waveguides of Arbitrary Cross-Section", Proc. IEE, Vol. 119, No.8, pp.1086-1091, August 1972.
- [14] P. B. Johns, "The Solution of Inhomogeneous Waveguide Problems Using a Transmission-Line Matrix", IEEE Trans. Microwave Theory Tech., Vol. MTT-22, No. 3, pp.209-215, March 1974.
- [15] S. Akhtarzad, and P. B. Johns, "Numerical Solution of Lossy Waveguides: T.L.M. Computer Program", Electron. Lett., Vol 10, No 15, pp.309-311, 25th July 1974.
- [16] P. B. Johns, "A New Mathematical Model to Describe the Physics of Propagation", The Radio and Electronic Engineer, Vol. 44, No. 12, pp.657-666, Dec. 1974.
- [17] S. Akhtarzad, "Analysis of Lossy Microwave Structures and Microstrip Resonators by the TLM Method", Ph.D. dissertation, University of Nottingham, England, July 1975.
- [18] P. Saguét, "Analyse des milieux guides - La method MTLM", Doctoral Thesis, Institut National Polytechnique de Grenoble, 1985.
- [19] P. Saguét, and E. Pic, "An Improvement for the TLM Method", Electron. Lett., Vol. 16, No. 7, pp.247-248, 27th March 1980.
- [20] Y.C. Shih, W.J.R. Hoefler, and A. Ros, "Cutoff Frequencies in Fin Lines Calculated with a Two-Dimensional TLM Program", in IEEE-MTT Intl. Microwave Symp. Digest, Whashington, D.C., pp.261-263, June 1980.
- [21] W. J. R. Hoefler and Y.-C. Shih., "Field Configuration of Fundamental and Higher Order Modes in Fin Lines obtained with the TLM Method", presented at URSI and Intl. IEEE-AP Symposium., Quebec, Canada., June 2-6. 1980.
- [22] Y. C. Shih, and W. J. R. Hoefler. "Dominant and Second-Order Mode Cutoff Frequencies in Fin Lines Calculated with a Two-Dimensional TLM Program". IEEE Trans. Microwave Theory Tech., Vol. MTT-28. No. 12. pp. 1443-1418. Dec. 1980.
- [23] Y. C. Shih, "The Analysis of Fin Lines Using Transmission Line Matrix and Transverse Resonance Methods", M.A.Sc. Thesis. University of Ottawa. Canada. 1980.
- [24] W. J. R. Hoefler, "The Transmission-Line Matrix Method - Theory and Applications", IEEE Trans. Microwave Theory Tech.. Vol. MTT-33. No. 10. pp. 882-893, Oct. 1985.
- [25] G. Kron, "Diakoptics", London: MacDonald, 1963. G. Kron. "Diakoptics", MacDonal. 1963.
- [26] A. Braemellar, M. N. John, M. R. Scott, "Practical Diakoptics for Electrical Networks", Chapman and Hall, London, 1969.
- [27] H. H. Happ, "Diakoptics and Networks" Academic Press, New York and London, 1971.
- [28] C. R. Brewitt-Tavior, and P.B. Johns, "On the Construction and Numerical Solution of Transmission-Line and Lumped Network Models of Maxwell's Equations", Int. J. Num. Meth. in Eng., Vol. 15, pp.13-30, 1980.

- [29] P. B. Johns, S. Akhtarzad, "The Use of Time Domain Diakoptics in Time Discrete Models of Fields", *Int. J. Numer. Methods Eng.*, Vol. 17, pp. 1-14, 1981.
- [30] P. B. Johns, S. Akhtarzad, "Time-domain approximations in the solution of fields by time domain diakoptics", *Int. J. Numerical Methods in Engineering*, vol. 18, pp. 1361-1373, 1982.
- [31] W. J. R. Hoefler, "Linear and Non-Linear Electromagnetic Field Modelling in the Time Domain with Transmission Line Modeling (TLM) Techniques", *MIOP' 89*, Feb. 28 - Mar. 2. 1989, Sindelfingen, Germany.
- [32] P. P. M. So, W.J.R. Hoefler, "General Planar Circuit Simulator based on Two-dimensional TLM Method", in *1989 IEEE Intl. Microwave Symp. Dig.*, June 12-16, 1989, Long Beach, CA.
- [33] W. J. R. Hoefler, "Linear and Nonlinear Modeling in the Time Domain with the Transmission Line Matrix (TLM) Method", *Alta Frequenza*, Vol. LVIII, No. 5-6, pp. 541-549 (105-113), Oct. 1989.
- [34] P. P. M. So, Eswarappa, W.J.R. Hoefler, "A Two-dimensional TLM Microwave Field Simulator using New Concepts and Procedures", *IEEE Trans. Microwave Theory Techniques*, Vol. MTT-37, No. 12, pp. 1877-1884, Dec. 1989.
- [35] Eswarappa, G. Costache, W. J. R. Hoefler, "TLM Modeling of Dispersive Wideband Absorbing Boundaries with Time Domain Diakoptics for S-Parameter Extraction", *IEEE Trans. Microwave Theory Techniques*, Vol. MTT-38, No. 4, pp. 379-386, April 1990.
- [36] W. J. R. Hoefler, "The Discrete Time Domain Green's Function or Johns Matrix - A New Powerful Concept in TLM", *Intl. Journal of Numerical Modeling*, Vol. 2, No. 4, pp. 215-225, 1989.
- [37] R. Simon, R. W. John and V. D. Theodore, "Fields and Waves in Communication Electronics", John Wiley Sons, INC, Toronto, 1994.
- [38] L. R. A. X. De Menezes e R. De Pádua Moreira, "Síntese de Filtros de Microondas Utilizando o Método TLM (Transmission Line Method)", *Anais do IV congresso Brasileiro de Eletromagnetismo*, Vol. 1, pp 220-223, Natal, 2000.
- [39] Dominguez. Hugo A., and Raizer. Adroaldo, "Using TLM Method to Model Hyperthermia Applicators", *Journal of Microwaves and Optoelectronics*, Vol. 2, N.o 6, December 2002.
- [40] Sadiku, Matthew N.O.. "Numerical Techniques in Electromagnetics", CRC Press, Florida, USA, pp. 51-523, 2000.
- [41] M. Pozar, David, "Microwave Engineering", Adisson Wesley, University of Massachusetts, 1990.
- [42] Saguet, P. and E. Pic., "Le maillage rectangulaire et le changement de maile dans la method TLM en deux dimensions", *Electronic Letters* 17, 277-278, 1981.

- [43] Al-Mukhtar, D.A. and J.E. Sich, "Transmission-line matrix method and irregularly graded space", Proc. of IEE Pt H 128, 299-305, 1981
- [44] Saguet, P. and E. Pic., "Utilisation d'un nouveau type de noeud dans la method TLM en 3 dimensions", Electronic Letters 18, 478-480, 1982.
- [45] Scaramuzza, R. A. and A. J. Lwery, "Hybrid symmetrical condensed node for the TLM method", Electronic Letters 26, 1947-1949, 1990.
- [46] Akhtarzad, S., and P. B. Johns., "Solution of Maxwell's equations in three space dimensions and time by the TLM method of numerical analysis. Proc. of IEE 122, 1344-1348, 1975."
- [47] P. B. Johns, "New symmetrical condensed node for three-dimensional solution of electromagnetic wave problems by TLM", Electronics Letters 22, 162-164, 1986.
- [48] P. B. Johns, "A symmetrical condensed node for the TLM", , IEEE Trans. on Microwave Theory and Tech., vol. MT-35, no. 4, pp.370-377, Apr. 1987.
- [49] J. L. Herring, and C. Christopoulos, "The application of different meshing techniques to EMC problems", Proceedings of the 9th Annual Review of Progress in Applied Electromagnetics, pp. 752-762, 1993.
- [50] Collin R. E. "Foundation for microwave engineering", New York, McGraw Hill, 1966.
- [51] C. A. Tenório De Carvalho, Jr., L. R. A. X. De Menezes and V. H. C. Melo, "Accelerating bidimensional TLM using adjustable time steps", Electronics Letters, Vol. 39 No. 25, 11th December 2003.
- [52] Poman P. M. So, Eswarappa and Wolfgang J. R. Hoefer, "A Two-Dimensional Transmission Line Matrix Microwave Field Simulator Using New Concepts and Procedures", IEEE Transaction On Microwave Theory and Techniques, Vol. 37, No. 12, December 1989.
- [53] Roger Horn and Charles Johnson. "Topics in Matrix Analysis", Chapter 4. Cambridge University Press, 1991.
- [54] Denecker B., Olyslager F., Knockaert, L. and De Zutter, D., "Generation of FDTD Subcell Equations by Means of Reduced Order Modeling", IEEE Transactions on Antennas and Propagation, Vol. 51, N° 8, August 2003.
- [55] Grivet-Talocia, S., "Package Macromodeling via Time-Domain Vector Fitting", IEEE Microwave Wireless Comp. Lett., vol. 13, No. 11, pp. 472-474, Nov. 2003.
- [56] Carvalho Júnior, Carlos Alberto Tenório de, MENEZES, Leonardo R A X de, MELO, Victor Hugo C. "High Performance TLM Simulation Using Adjustable Time Steps". International Microwave Symposium IEEE MTT-S, Forth Worth, Texas, June 2004.
- [57] Yuanxun Wang and Tatsuo Itoh, "Envelope-finite element (EVFE) technique-2-D guided wave examples", IEEE MTT-S, WE1EF-1, pp. 741-744, 2001.
- [58] Yuanxun Wang and Tatsuo Itoh, "Envelope-finite element (EVFE) technique – a more efficient time-domains scheme", IEEE Transaction On Microwave Theory and Techniques, Vol. 49, No. 12, pp. 2241-2247, Dec., 2001.

- [59] Hsiao-Ping Tsai, Yuanxun Wang and Tatsuo Itoh, "Efficient Analysis of Microwave Passive Structures Using 3-D Envelope-Finite Element (EVFE)", IEEE Transaction On Microwave Theory and Techniques, Vol. 50, No. 12, pp. 2721-2727, Dec., 2002.
- [60] Shu-Hai Sun and Charles T. M. Choi, "Performance of the Improved PML for the Envelope ADI-FDTD Method in Two-Dimensional Domain", IEEE Microwave Wireless Comp. Lett., vol. 15, No. 11, pp. 820-822, Nov., 2005.
- [61] Charles T. M. Choi and Shu-Hai Sun, "Numerical Performance and Applications of the Envelope ADI FDTD Method", IEEE Transaction On Microwave Theory and Techniques, Vol. 54, No. 1, pp. 256-264, Jan. 2006.
- [62] C.-H. Lai, "Diakoptics, Domain Decomposition and Parallel Computing", The Computer Journal, Volume 37, Issue 10, pp. 840-846, 1972.
- [63] A. Brameller, M. N. John and M. R. Scott, "Practical Diakoptics for Eletrical Networks", Chapman and Hall, London, 1969.
- [64] H. H. Happ, "Diakoptics and Networks", Academic Press, New York and London, 1971.
- [65] O. C. Zeienkiewicz, "The Finite Element Method", McGraw-Hill, London, 1977.
- [66] F. W. Beaufait, W. H. Rowan, P.G. Hoadley and R. M. Hackett, "Computer Methods of Structural Analysis", Prentice-Hall, Englewood Cliffs, N.J., 1970.
- [67] T. Okoshi, Y. Uehara, and T. Takeuchi, "The segmentation method –An approach to the analysis of microwave planar circuits", IEEE Trans. Microwave Theory Tech., vol MTT-24, pp 662-668,1976.
- [68] R. Chadha and K. C. Gupta, "Segmentation method using impedance matrices for analysis of planar microwave circuits", IEEE Trans. Microwave Theory Tech., vol. MTT-29, pp. 71-74, 1981.
- [69] R. Sorrentino, "Planar circuits, waveguide models and segmentation method", IEEE Trans. Microwave Theory Tech., vol. MTT-33, pp. 1057-1066, Oct. 1985.
- [70] P. B. Johns and K. Akhtarzad, "The use of time domains diakoptics in time discrete models of fields", Int. J. Numer. Methods Eng., vol.17, pp.1-14, 1981.
- [71] P. B. Johns and K. Akhtarzad, "Time domain approximations in the solution of fields by time domain diakoptics", Int. J. Numer. Methods Eng., vol.18, pp.1361-1373, 1982.

APÊNDICES

A– FERRAMENTA E CÓDIGO USADO NA PESQUISA

A.1– PRODUTO TENSORIAL DE KRONNECKER NA OBTENÇÃO DA MATRIZ DE ESPALHAMENTO

A.1.1– Definição

O produto tensorial de Kronecker (\otimes) é definido da seguinte forma:
Dadas as matrizes $A_{m \times n}$ e $B_{p \times q}$ o produto será,

$$A_{m \times n} \otimes B_{p \times q} \hat{=} (a_{ij} B)_{mp \times pq} \quad (\text{A.1})$$

Sempre que A_1 e B_1 possuírem a mesma dimensão e A_2, B_2 idem,

$$(A_1 \otimes B_1) \cdot (A_2 \otimes B_2) = A_1 \cdot A_2 \otimes B_1 \cdot B_2 \quad (\text{A.2})$$

A.1.2– Decompondo a Matriz S em Produto Tensorial de Kronecker

Dada a matriz S ,

$$S = \begin{bmatrix} M & N \\ N & M \end{bmatrix} = \begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix} + \begin{bmatrix} 0 & N \\ N & 0 \end{bmatrix} = I_2 \otimes M + P_2 \otimes N \quad (\text{A.3})$$

onde

$$I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (\text{A.4})$$

e

$$P_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (\text{A.5})$$

Vamos explodir a matriz M .

$$M = \left[\begin{array}{cc|ccc} -1 & 1 & \dots & 0 & 0 \\ 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \hline 0 & 0 & \dots & -1 & 1 \\ 0 & 0 & \dots & 1 & -1 \end{array} \right] = I_{2n} \otimes K \quad (\text{A.6})$$

e de forma similar,

$$N = I_{2n} \otimes U \quad (\text{A.7})$$

onde

$$K = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \iff K^w = (-2)^{w-1} \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \quad (\text{A.8})$$

$$U = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \iff U^w = 2^{w-1} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad (\text{A.9})$$

Com isso,

$$S = I_2 \otimes I_{2n} \otimes K + P_2 \otimes I_{2n} \otimes U \quad (\text{A.10})$$

$$S_A = I_2 \otimes I_{2n} \otimes K$$

$$S_B = P_2 \otimes I_{2n} \otimes U$$

Note que S_A e S_B são comutativas! Basta verificar que $KU = UK$, o que é verdadeiro. Agora o processo pode ser facilmente diagonalizado.

Teorema. Se A e B são quadradas.

$$AV_A = V_A \cdot \Lambda_A$$

$$BV_B = V_B \cdot \Lambda_B$$

onde

Λ_A, Λ_B estão na forma de Jordan.

V_A, V_B são matrizes dos autovetores.

$$\begin{aligned} A \otimes B &= V_A \cdot \Lambda_A \cdot V_A^{-1} \otimes V_B \cdot \Lambda_B \cdot V_B^{-1} \\ &= (V_A \otimes V_B) \cdot (\Lambda_A \cdot V_A^{-1} \otimes \Lambda_B \cdot V_B^{-1}) \\ &= (V_A \otimes V_B) \cdot (\Lambda_A \otimes \Lambda_B) \cdot (V_A^{-1} \otimes V_B^{-1}) \\ &= V_{AB} \cdot \Lambda_{AB} \cdot V_{AB}^{-1} \end{aligned} \quad (\text{A.11})$$

$$(S_A + S_B)^2 = (I_2 \otimes I_{2n} \otimes K)^2 + S_A \cdot S_B + S_B \cdot S_A + (P_2 \otimes I_{2n} \otimes U)^2 \quad (\text{A.12})$$

Mas $S_A \cdot S_B = 0$

A.1.3– Proposição

Em geral

$$(S_A + S_B)^t = I_2 \otimes I_{2n} \otimes K^t + P_2 \otimes I_{2n} \otimes U^t \quad (\text{A.13})$$

Nota-se que

$$K^t = (-2)^{t-1} \cdot K$$

$P^t = I$ se t for par e $P^t = P$ se t for ímpar.

$$U^t = 2^{t-1} \cdot U$$

Sempre que t for par

$$(S_A + S_B)^t = 2^{t-1} (S_A + S_B) \quad (\text{A.14})$$

Processo de Diagonalização

$$P_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad (\text{A.15})$$

$$V_{P_2} = Q = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (\text{A.16})$$

$$V_{P_2}^{-1} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (\text{A.17})$$

$$V_K = V_U = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (\text{A.18})$$

$$V_K^{-1} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (\text{A.19})$$

$$V_{P2} = Q = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (\text{A.20})$$

$$8S = (Q \otimes I_{2n} \otimes Q) \cdot \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes I_{2n} \otimes \begin{bmatrix} 0 & 0 \\ 0 & -2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \otimes I_{2n} \otimes \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \right) \cdot (Q \otimes I_{2n} \otimes Q) \quad (\text{A.21})$$

A.2– CÓDIGO EM MATLAB DO TLM-JSN BIDIMENSIONAL

Nesta seção apresenta-se os códigos utilizados nas simulações feitas para testes de convergência do método JSN (JSN puro, JSN ROM 1 e JSN ROM 2). Mostra-se também, os códigos que geram as estruturas simuladas: Iris 1, 2 e 3.

A.2.1– IRIS 1

```
clear
% Tamanho da Regiao a ser modelada
m1=10;
m2=40;
% Cria a matriz que descreve se ha elementos metalicos na regioa modelada
A3=zeros(2*m1+1,2*m2+1);
% Preenche a matriz de elementos metalicos (-1), magneticos (1) ou vazio (10)

for k=1:m1
    for l=1:m2
        A3(2*k,2*l)=10;
        A3(1,2*l)=1;
        A3(2*m1+1,2*l)=1;
        A3(2*k,2*m2+1)=0;
    end;
end;
for k=2:8
    A3(11,2*k)=1;
end;
for k=12:18
    A3(11,2*k)=1;
end;
for k=22:28
    A3(11,2*k)=1;
end;
for k=32:38
    A3(11,2*k)=1;
end;
```

A.2.2– IRIS 2

```
clear
% Tamanho da Regiao a ser modelada
m1=10;
m2=50;
% Cria a matriz que descreve se ha elementos metalicos na regioa modelada
A3=zeros(2*m1+1,2*m2+1);
% Preenche a matriz de elementos metalicos (-1), magneticos (1) ou vazio (10)
for k=1:m1
```

```

    for l=1:m2
        A3(2*k,2*1)=10;
        A3(1,2*1)=1;
        A3(2*m1+1,2*1)=1;
        A3(2*k,2*m2+1)=0;
    end;
end;
for l=1:6
    A3(7,2*1)=1;
    A3(13,2*1)=1;
end;
for l=12:24
    A3(7,2*1)=1;
    A3(13,2*1)=1;
end;
for l=30:40
    A3(7,2*1)=1;
    A3(13,2*1)=1;
end;
for l=46:50
    A3(7,2*1)=1;
    A3(13,2*1)=1;
end;
for k=1:3
    A3(2*k,13)=1;
    A3(2*k,23)=1;
    A3(2*k,49)=1;
    A3(2*k,59)=1;
    A3(2*k,81)=1;
    A3(2*k,91)=1;
end;
for k=7:10
    A3(2*k,13)=1;
    A3(2*k,23)=1;
    A3(2*k,49)=1;
    A3(2*k,59)=1;
    A3(2*k,81)=1;
    A3(2*k,91)=1;
end;

```

A.2.3- IRIS 3

```

clear
% Tamanho da Regiao a ser modelada
m1=10;
m2=40;
% Cria a matriz que descreve se ha elementos metalicos na regiao modelada
A3=zeros(2*m1+1,2*m2+1);
% Preenche a matriz de elementos metalicos (-1), magneticos (1) ou vazio (10)

for k=1:m1
    for l=1:m2
        A3(2*k,2*1)=10;
        A3(1,2*1)=1;
        A3(2*m1+1,2*1)=1;
        A3(2*k,2*m2+1)=0;
    end;
end;
for k=1:m1
    A3(2*k,21)=1;
    A3(2*k,43)=1;
    A3(2*k,63)=1;
end;
for k=4:7
    A3(2*k,21)=0;
    A3(2*k,43)=0;
    A3(2*k,63)=0;
end;

```

A.2.4- JSN PURO

```

% Coloque aqui o nome do arquivo de estruturas

```

```

run iris1
A=ones(m1,m2);
% Completa a matriz a ser modelada com numeros inteiros crescentes
x1=0;
for k=1:m1;
    for l=1:m2
        A(k,l)=1+x1;
        x1=A(k,l);
    end;
end;

% Cria matrizes auxiliares para o calculo das matrizes T, S e C
% Cria matrizes auxiliares para o calculo de C
B1=zeros(A(m1,m2),A(m1,m2));
B2=zeros(A(m1,m2),A(m1,m2));
B3=zeros(A(m1,m2),A(m1,m2));
B4=zeros(A(m1,m2),A(m1,m2));
B11=zeros(A(m1,m2),A(m1,m2));
B22=zeros(A(m1,m2),A(m1,m2));
B33=zeros(A(m1,m2),A(m1,m2));
B44=zeros(A(m1,m2),A(m1,m2));
Cz=zeros(4,4);
C1=Cz;
C2=Cz;
C11=Cz;
C22=Cz;
C33=Cz;
C44=Cz;
% Matrizes para quando o no esta proximo a uma parede eletrica ou magnetica
C11(1,1)=1;
C22(2,2)=1;
C33(3,3)=1;
C44(4,4)=1;
% Matrizes para quando o no nao esta proximo a paredes
C1(1,3)=1;
C2(2,4)=1;
C1T=C1';
C2T=C2';
szx=size(B1);
szx1=szx(1);
szx2=szx(2);
% Preenche as submatrizes auxiliares
for k=1:m1
    for l=1:m2
        if l>1
            if A3(2*k,2*l-1)==0
                B1(A(k,l),A(k,l-1))=1;
            else
                B11(A(k,l),A(k,l))=A3(2*k,2*l-1);
            end;
        elseif l==1
            B11(A(k,l),A(k,l))=A3(2*k,1);
        end;
        if k<m1
            if A3(2*k+1,2*l)==0
                B2(A(k,l),A(k+1,l))=1;
            else
                B22(A(k,l),A(k,l))=A3(2*k+1,2*l);
            end;
        elseif k==m1
            B22(A(k,l),A(k,l))=A3(2*m1+1,2*l);
        end;
        if k>1
            if A3(2*k-1,2*l)==0
                B3(A(k,l),A(k-1,l))=1;
            else
                B33(A(k,l),A(k,l))=A3(2*k-1,2*l);
            end;
        elseif k==1
            B33(A(k,l),A(k,l))=A3(1,2*l);
        end;
        if l<m2
            if A3(2*k,2*l+1)==0

```

```

        B4(A(k,l),A(k,l+1))=1;
    else
        B44(A(k,l),A(k,l))=A3(2*k,2*m+1);
    end;
elseif l==m2
    B44(A(k,l),A(k,l))=A3(2*k,2*m2+1);
end;
end;
end;
% Utiliza Esparsidade
B1=sparse(B1);
B11=sparse(B11);
B2=sparse(B2);
B22=sparse(B22);
B3=sparse(B3);
B33=sparse(B33);
B4=sparse(B4);
B44=sparse(B44);
C1=sparse(C1);
C1T=sparse(C1T);
C2=sparse(C2);
C2T=sparse(C2T);
C11=sparse(C11);
C22=sparse(C22);
C33=sparse(C33);
C44=sparse(C44);
disp('Preenche matrizes');
% Cria a matriz final de conexao C
C=kron(B1,C2T)+kron(B2,C1T)+kron(B3,C1)+kron(B4,C2)+kron(B11,C44)+kron(B22,C33)+kron(B33,C11)+kron(B44,C22);
% Cria matrizes auxiliares para o calculo da matriz final S
S1=sparse(eye(szx1,szx2));
S2=sparse(1/2*ones(4,4)-eye(4,4));
% Calcula a matriz S final
S=kron(S1,S2);
% Cria a matriz externa auxiliar para o calculo de T
A2=zeros(m1+2,m2+2);
% Preenche a matriz externa auxiliar para calcular T
x1=0;
for k=1:m1+2;
    A2(k,1)=1+x1;
    x1=A2(k,1);
end;
for k=1:m1+2;
    A2(k,m2+2)=1+x1;
    x1=A2(k,m2+2);
end;
for l=2:m2+1
    A2(1,l)=1+x1;
    x1=A2(1,l);
end;
for l=2:m2+1
    A2(m1+2,l)=1+x1;
    x1=A2(m1+2,l);
end;
% Cria matrizes auxiliares para o calculo de T
T1=zeros(2*(m1+m2+4)-4,m1*m2);
T2=zeros(2*(m1+m2+4)-4,m1*m2);
T3=zeros(2*(m1+m2+4)-4,m1*m2);
T4=zeros(2*(m1+m2+4)-4,m1*m2);
% Preenche as matrizes auxiliares com os dados de A e de A2
for l=2:m2+1
    if A3(1,2*m1-2)==0
        T3(A2(1,l),A(1,l-1))=1;
    else
        T3(A2(1,l),A(1,l-1))=0;
    end;
    if A3(2*m1+1,2*m1-2)==0
        T2(A2(m1+2,l),A(m1,l-1))=1;
    else
        T2(A2(m1+2,l),A(m1,l-1))=0;
    end;
end;
end;
% Fecha as portas cobertas por paredes utilizando os dados de A3

```

```

for k=2:m1+1
    if A3(2*k-2,1)==0
        T1(A2(k,1),A(k-1,1))=1;
    else
        T1(A2(k,1),A(k-1,1))=0;
    end;
    if A3(2*k-2,2*m2+1)==0
        T4(A2(k,m2+2),A(k-1,m2))=1;
    else
        T4(A2(k,m2+2),A(k-1,m2))=0;
    end;
end;
% Calcula as matrizes auxiliares C?x
C1x=C1(1,:);
C2x=C2(2,:);
C3x=C1T(3,:);
C4x=C2T(4,:);
% Mais esparsidade para o calculo de T
T1=sparse(T1);
T2=sparse(T2);
T3=sparse(T3);
T4=sparse(T4);
% Calcula a matriz T final
T=kron(T1,C2x)+kron(T2,C1x)+kron(T3,C3x)+kron(T4,C4x);
% Inicio de simulacoes
% Limpa variaveis auxiliares da memoria
clear B1 B2 B3 B4 T1 T2 T3 T4 C1 C2 S1 S2 B11 B22 B33 B44 C1T C2T C1x C2x C3x C4x Cz C11 C22 C33 C44
%clear A A1 A2 A3
% Define numeros de passos de tempo
disp('Inicia calculos')
NT=2000;
% Inicializa os vetores incidentes externos (vei), internos (vii) e refetidos (ver)
ver=zeros(2*(m1+m2+4)-4,1);
vei=zeros(2*(m1+m2+4)-4,1);
vii=ones(4*m1*m2,1);
% Inicializa o vetor incidente interno para o TLM rapido
vi2=ones(4*m1*m2,1);
% Inicializa vetor de saida vol
vol=sparse(zeros(2*(m1+m2+4)-4,NT));
% Inicializa vetor gaussiano
vx1=zeros(1,NT);
% Calcula as matrizes Ax e Bx de y[k]=Ax*w[k] e w[k]=T'*x[k]+Bx*w[k-1]
Ax=sparse(T*S);
Bx=sparse(C*S);
% Inicializa o vetor de excitacao para t=0
vei(4)=exp(-120*((NT/3)/(NT/8))^2);
vx1(1)=vei(4);
% Calcula w[0] e y[0]
vii=T'*vei;
ver=Ax*vii;
U=sparse(eye(4*m1*m2,4*m1*m2));
disp('Inicia TLM lento')
% Calcula vetor de saida para t=0
vol(:,1)=ver;
% Inicia calculo para t>0
for k=1:NT-1;
    vei(4)=exp(-120*((k-NT/3)/(NT/8))^2);
    vx1(k+1)=vei(4);
    vii=T'*vei+Bx*vii;
    ver=Ax*vii;
    vol(:,k+1)=ver+T*vii;
end;
disp('Milhoes de operacoes no TLM lento')
ox1=1e-6*NT*((4*m1*m2*(nnz(Bx)+nnz(Ax))+(2*(m1+m2+4)-4)*nnz(T)));
disp(ox1);
disp('Inicia matrizes do TLM rapido')
% Inicializa matrizes para o TLM rapido
Ts1=U;
Ts2=0*U;
Bsx=Bx;
% Define a razao entre o passo de tempo rapido e o lento
p=2;
% Calcula parte das matrizes do TLM rapido

```

```

disp('Calculo da matriz Ts1, Ts2 e Bsx')
for l=1:p-1;
    Ts1=U+Bx*Ts1;
    Ts2=Ts2+l*Bsx;
    Bsx=Bx*Bsx;
end;
disp('Calculo da matriz Ts2')
% Calcula as matrizes restantes do TLM rapido
Tsx=Ts1*T';
Tsy=Ts2*T';
% Limpa as matrizes nao esparsas Ts1, Ts2 e U
clear Ts1 Ts2 U
% Numero de passos de tempo do TLM rapido
NT1=NT/p;
% Inicializa vetor de saida vo2
vo2=zeros(2*(m1+m2+4)-4,NT1);
% Inicializa vetor gaussiano
vx2=zeros(1,NT1);
vx2=vei(4);
% Calcula w[0] e saida para o TLM rapido
vi2=T'*vei;
vo2(:,1)=Ax*vi2;
% Inicia algoritimo de calculo do TLM rapido
disp('Inicia TLM rapido')
for k=1:1:NT1-1;
    xx=vei;
    vei(4)=exp(-120*((p*k-NT/3)/(NT/8))^2);
    vx2(k+1)=vei(4);
    vi2=Tsx*vei+Tsy*(xx-vei)/p+Bsx*vi2;
    vo2(:,k+1)=Ax*vi2+T*vi2;
end;
disp('Milhoes de operacoes no TLM rapido')
ox2=1e-6*NT1*((4*m1*m2*(nnz(Bsx)+nnz(Ax)))+(2*(m1+m2+4)-4)*(nnz(Tsx)+nnz(Tsy)));
disp(ox2);
% Calculo dos vetores de saida
y1=vo1(33,:);
y2=vo2(33,:);
% Saida grafica do sistema
disp('Saida Grafica')
figure
% Dominio do tempo
plot(1:NT,y1,1:p*NT1,y2,'-+')
xlabel('Passos de Tempo');
ylabel('Amplitude');
title('TLM rapido p=5 - Dominio do Tempo');
% Dominio da Frequencia
h1=fft(full(y1));
h2=fft(full(y2));
h01=fft(vx1);
h02=fft(vx2);
figure
f1=(0:(NT-1)/10)/(NT-1);
f2=(0:(NT1-1)/10)/(p*(NT1-1));
plot(f1,abs(h1(1:round(NT/10))),f2,abs(p*h2(1:round(NT1/10))),'+-')
xlabel('Passos de Tempo');
ylabel('Magnitude');
title('TLM rapido p=5 - Dominio da Frequencia');
figure
plot(f1,180/pi*angle(h1(1:round(NT/10))),f2,180/pi*angle(p*h2(1:round(NT1/10))),'+-')
xlabel('Passos de Tempo');
ylabel('Fase');
title('TLM rapido p=5 - Dominio da Frequencia');

```

A.2.5- JSN ROM 1

```

% Coloque aqui o nome do arquivo de estruturas
run iris1
% Matriz que descreve a regioa a ser modelada
A=ones(m1,m2);
% Completa a matriz a ser modelada com numeros inteiros crescentes
x1=0;
for k=1:m1;
    for l=1:m2

```

```

        A(k,l)=1+x1;
        x1=A(k,l);
    end;
end;

% Cria matrizes auxiliares para o calculo das matrizes T, S e C
% Cria matrizes auxiliares para o calculo de C
B1=zeros(A(m1,m2),A(m1,m2));
B2=zeros(A(m1,m2),A(m1,m2));
B3=zeros(A(m1,m2),A(m1,m2));
B4=zeros(A(m1,m2),A(m1,m2));
B11=zeros(A(m1,m2),A(m1,m2));
B22=zeros(A(m1,m2),A(m1,m2));
B33=zeros(A(m1,m2),A(m1,m2));
B44=zeros(A(m1,m2),A(m1,m2));
Cz=zeros(4,4);
C1=Cz;
C2=Cz;
C11=Cz;
C22=Cz;
C33=Cz;
C44=Cz;
% Matrizes para quando o no esta proximo a uma parede eletrica ou magnetica
C11(1,1)=1;
C22(2,2)=1;
C33(3,3)=1;
C44(4,4)=1;
% Matrizes para quando o no nao esta proximo a paredes
C1(1,3)=1;
C2(2,4)=1;
C1T=C1';
C2T=C2';
szx=size(B1);
szx1=szx(1);
szx2=szx(2);
% Preenche as submatrizes auxiliares
for k=1:m1
    for l=1:m2
        if l>1
            if A3(2*k,2*l-1)==0
                B1(A(k,l),A(k,l-1))=1;
            else
                B11(A(k,l),A(k,l))=A3(2*k,2*l-1);
            end;
        elseif l==1
            B11(A(k,l),A(k,l))=A3(2*k,1);
        end;
        if k<m1
            if A3(2*k+1,2*l)==0
                B2(A(k,l),A(k+1,l))=1;
            else
                B22(A(k,l),A(k,l))=A3(2*k+1,2*l);
            end;
        elseif k==m1
            B22(A(k,l),A(k,l))=A3(2*m1+1,2*l);
        end;
        if k>1
            if A3(2*k-1,2*l)==0
                B3(A(k,l),A(k-1,l))=1;
            else
                B33(A(k,l),A(k,l))=A3(2*k-1,2*l);
            end;
        elseif k==1
            B33(A(k,l),A(k,l))=A3(1,2*l);
        end;
        if l<m2
            if A3(2*k,2*l+1)==0
                B4(A(k,l),A(k,l+1))=1;
            else
                B44(A(k,l),A(k,l))=A3(2*k,2*l+1);
            end;
        elseif l==m2
            B44(A(k,l),A(k,l))=A3(2*k,2*m2+1);
        end;
    end;
end;

```

```

        end;
    end;
end;
% Utiliza Esparsidade
B1=sparse(B1);
B11=sparse(B11);
B2=sparse(B2);
B22=sparse(B22);
B3=sparse(B3);
B33=sparse(B33);
B4=sparse(B4);
B44=sparse(B44);
C1=sparse(C1);
C1T=sparse(C1T);
C2=sparse(C2);
C2T=sparse(C2T);
C11=sparse(C11);
C22=sparse(C22);
C33=sparse(C33);
C44=sparse(C44);
disp('Preenche matrizes');
% Cria a matriz final de conexao C
C=kron(B1,C2T)+kron(B2,C1T)+kron(B3,C1)+kron(B4,C2)+kron(B11,C44)+kron(B22,C33)+kron(B33,C11)+kron(B44,C22);
% Cria matrizes auxiliares para o calculo da matriz final S
S1=sparse(eye(szx1,szx2));
S2=sparse(1/2*ones(4,4)-eye(4,4));
% Calcula a matriz S final
S=kron(S1,S2);
% Cria a matriz externa auxiliar para o calculo de T
A2=zeros(m1+2,m2+2);
% Preenche a matriz externa auxiliar para calcular T
x1=0;
for k=1:m1+2;
A2(k,1)=1+x1;
x1=A2(k,1);
end;
for k=1:m1+2;
A2(k,m2+2)=1+x1;
x1=A2(k,m2+2);
end;
for l=2:m2+1
A2(1,l)=1+x1;
x1=A2(1,l);
end;
for l=2:m2+1
A2(m1+2,l)=1+x1;
x1=A2(m1+2,l);
end;
% Cria matrizes auxiliares para o calculo de T
T1=zeros(2*(m1+m2+4)-4,m1*m2);
T2=zeros(2*(m1+m2+4)-4,m1*m2);
T3=zeros(2*(m1+m2+4)-4,m1*m2);
T4=zeros(2*(m1+m2+4)-4,m1*m2);
% Preenche as matrizes auxiliares com os dados de A e de A2
for l=2:m2+1
    if A3(1,2*l-2)==0
        T3(A2(1,l),A(1,l-1))=1;
    else
        T3(A2(1,l),A(1,l-1))=0;
    end;
    if A3(2*m1+1,2*l-2)==0
        T2(A2(m1+2,l),A(m1,l-1))=1;
    else
        T2(A2(m1+2,l),A(m1,l-1))=0;
    end;
end;
end;
% Fecha as portas cobertas por paredes utilizando os dados de A3
for k=2:m1+1
    if A3(2*k-2,1)==0
        T1(A2(k,1),A(k-1,1))=1;
    else
        T1(A2(k,1),A(k-1,1))=0;
    end;
end;

```

```

    if A3(2*k-2,2*m2+1)==0
        T4(A2(k,m2+2),A(k-1,m2))=1;
    else
        T4(A2(k,m2+2),A(k-1,m2))=0;
    end;
end;
% Calcula as matrizes auxiliares C?x
C1x=C1(1,:);
C2x=C2(2,:);
C3x=C1T(3,:);
C4x=C2T(4,:);
% Mais esparsidade para o calculo de T
T1=sparse(T1);
T2=sparse(T2);
T3=sparse(T3);
T4=sparse(T4);
% Calcula a matriz T final
T=kron(T1,C2x)+kron(T2,C1x)+kron(T3,C3x)+kron(T4,C4x);
% Inicio de simulacoes
% Limpa variaveis auxiliares da memoria
clear B1 B2 B3 B4 T1 T2 T3 T4 C1 C2 S1 S2 B11 B22 B33 B44 C1T C2T C1x C2x C3x C4x Cz C11 C22 C33 C44
%clear A A1 A2 A3
% Define numeros de passos de tempo
disp('Inicia calculos')
NT=1500;
% Inicializa os vetores incidentes externos (vei), internos (vii) e refletidos (ver)
ver=zeros(2*(m1+m2+4)-4,1);
vei=zeros(2*(m1+m2+4)-4,1);
vii=ones(4*m1*m2,1);
% Inicializa o vetor incidente interno para o TLM rapido
%vi2=ones(4*m1*m2,1);
% Inicializa vetor de saida vo1
vo1=sparse(zeros(2*(m1+m2+4)-4,NT));
% Inicializa vetor gaussiano
vx1=zeros(1,NT);
% Calcula as matrizes Ax e Bx de y[k]=Ax*w[k] e w[k]=T'*x[k]+Bx*w[k-1]
Ax=sparse(T*S);
Bx=sparse(C*S);
% Inicializa o vetor de excitacao para t=0
vei(6)=exp(-100*((NT/2)/(NT/8))^2);
vx1(1)=vei(6);
% Calcula w[0] e y[0]
vii=T'*vei;
ver=Ax*vii;
U=sparse(eye(4*m1*m2,4*m1*m2));
disp('Inicia TLM lento')
% Calcula vetor de saida para t=0
vo1(:,1)=ver;
% Inicia calculo para t>0
t0=clock;
for k=1:NT-1;
    vei(6)=exp(-100*((k-NT/2)/(NT/8))^2);
    vx1(k+1)=vei(6);
    vii=T'*vei+Bx*vii;
    ver=Ax*vii;
    vo1(:,k+1)=ver+T*vii;
end;
disp('Tempo no TLM lento')
%ox1=1e-6*NT*((4*m1*m2*(nnz(Bx)+nnz(Ax))+(2*(m1+m2+4)-4)*nnz(T)));
ox1=etime(clock,t0);
disp(ox1);
disp('Inicia matrizes do TLM rapido')
%q=2;
q=2*(m1*m2-m1-m2+2);
% Inicializa matrizes para o TLM rapido
[u1,t1]=schur(full(Bx));

Ts1=U;
Ts2=0*U;
Bx=sparse(t1);
Bsx=Bx;
% Define a razao entre o passo de tempo rapido e o lento
p=160;

```

```

% Calcula parte das matrizes do TLM rapido
disp('Calculo da matriz Ts1, Ts2 e Bsx')
for l=1:p-1;
    x=1*100/p
    disp('Porcentos Concluidos')
    Ts1=U+Bx*Ts1;
    Ts2=Ts2+l*Bsx;
    Bsx=Bx*Bsx;
end;
disp('Calculo da matriz Ts2')
% Calcula as matrizes restantes do TLM rapido
Tsx=sparse(Ts1*u1'*T');
Tsy=sparse(Ts2*u1'*T');
Ax=sparse(Ax*u1);
T=sparse(T*u1);
Tsx=sparse(Tsx(1:(4*m1*m2-q+1),1:(2*(m1+m2+4)-4)));
Tsy=sparse(Tsy(1:(4*m1*m2-q+1),1:(2*(m1+m2+4)-4)));
Bx=sparse(Bx(1:(4*m1*m2-q+1),1:(4*m1*m2-q+1)));
Bsx=sparse(Bsx(1:(4*m1*m2-q+1),1:(4*m1*m2-q+1)));
Ax=sparse(Ax(1:(2*(m1+m2+4)-4),1:(4*m1*m2-q+1)));
T=sparse(T(1:(2*(m1+m2+4)-4),1:(4*m1*m2-q+1)));
disp('Reducao de modelo - matriz antiga')
ox3=(4*m1*m2);
disp(ox3);
disp('Reducao de modelo - matriz nova')
ox3=(4*m1*m2-q+1);
disp(ox3);
% Limpa as matrizes nao esparsas Ts1, Ts2 e U
clear Ts1 Ts2 U
% Numero de passos de tempo do TLM rapido
NT1=NT/p;
% Inicializa vetor vi2
vi2=ones(4*m1*m2-q+1,1);
% Inicializa vetor de saida vo2
vo2=zeros(2*(m1+m2+4)-4,NT1);
% Inicializa vetor gaussiano
vx2=zeros(1,NT1);
vx2=vei(4);
% Calcula w[0] e saida para o TLM rapido
vi2=T'*vei;
vo2(:,1)=Ax*vi2;
% Inicia algoritimo de calculo do TLM rapido
disp('Inicia TLM rapido')
t0=clock;
for k=1:1:NT1-1;
    xx=vei;
    vei(6)=exp(-100*((p*k-NT/2)/(NT/8))-2);
    vx2(k+1)=vei(6);
    vi2=Tsx*vei+Tsy*(xx-vei)/p+Bsx*vi2;
    vo2(:,k+1)=Ax*vi2+T*vi2;
end;
disp('Tempo no TLM rapido')
%ox2=1e-6*NT1*(((4*m1*m2-q)*(nnz(Bsx)+nnz(Ax))+2*(m1+m2+4)-4)*(nnz(Tsx)+nnz(Tsy)));
ox2=etime(clock,t0);
disp(ox2);
% Calculo dos vetores de saida
y1=full(vo1(18,:));
y2=full(vo2(18,:));
% Saida grafica do sistema
disp('Saida Grafica')
figure
% Dominio do tempo
plot(1:NT,y1,1:p*NT1,y2,'-+')
xlabel('Passos de Tempo');
ylabel('Amplitude');
title('TLM rapido p=5 - Dominio do Tempo');
% Dominio da Frequencia
h1=fft(full(y1));
h2=fft(full(y2));
h01=fft(vx1);
h02=fft(vx2);
figure
f1=(0:(NT-1)/10)/(NT-1);

```

```

f2=(0:(NT1-1)/10)/(p*(NT1-1));
plot(f1,abs(h1(1:round(NT/10))),f2,abs(p*h2(1:round(NT1/10))),'+-')
xlabel('Passos de Tempo');
ylabel('Magnitude');
title('TLM rapido p=5 - Dominio da Frequencia');
figure
plot(f1,180/pi*angle(h1(1:round(NT/10))),f2,180/pi*angle(p*h2(1:round(NT1/10))),'+-')
xlabel('Passos de Tempo');
ylabel('Fase');
title('TLM rapido p=5 - Dominio da Frequencia');

```

A.2.6- JSN ROM 2

```

%Escreva aqui o nome do arquivo de estrutura
run iris1
% Matriz que descreve a regioa a ser modelada
A=ones(m1,m2);
% Completa a matriz a ser modelada com numeros inteiros crescentes
x1=0;
for k=1:m1;
    for l=1:m2
        A(k,l)=1+x1;
        x1=A(k,l);
    end;
end;
% Cria matrizes auxiliares para o calculo das matrizes T, S e C
% Cria matrizes auxiliares para o calculo de C
B1=zeros(A(m1,m2),A(m1,m2));
B2=zeros(A(m1,m2),A(m1,m2));
B3=zeros(A(m1,m2),A(m1,m2));
B4=zeros(A(m1,m2),A(m1,m2));
B11=zeros(A(m1,m2),A(m1,m2));
B22=zeros(A(m1,m2),A(m1,m2));
B33=zeros(A(m1,m2),A(m1,m2));
B44=zeros(A(m1,m2),A(m1,m2));
Cz=zeros(4,4);
C1=Cz;
C2=Cz;
C11=Cz;
C22=Cz;
C33=Cz;
C44=Cz;
% Matrizes para quando o no esta proximo a uma parede eletrica ou magnetica
C11(1,1)=1;
C22(2,2)=1;
C33(3,3)=1;
C44(4,4)=1;
% Matrizes para quando o no nao esta proximo a paredes
C1(1,3)=1;
C2(2,4)=1;
C1T=C1';
C2T=C2';
szx=size(B1);
szx1=szx(1);
szx2=szx(2);
% Preenche as submatrizes auxiliares
for k=1:m1
    for l=1:m2
        if l>1
            if A3(2*k,2*l-1)==0
                B1(A(k,l),A(k,l-1))=1;
            else
                B11(A(k,l),A(k,l))=A3(2*k,2*l-1);
            end;
        elseif l==1
            B11(A(k,l),A(k,l))=A3(2*k,1);
        end;
        if k<m1
            if A3(2*k+1,2*l)==0
                B2(A(k,l),A(k+1,l))=1;
            else
                B22(A(k,l),A(k,l))=A3(2*k+1,2*l);
            end;
        end;
    end;
end;

```

```

elseif k==m1
    B22(A(k,l),A(k,l))=A3(2*m1+1,2*1);
end;
if k>1
    if A3(2*k-1,2*1)==0
        B3(A(k,l),A(k-1,l))=1;
    else
        B33(A(k,l),A(k,l))=A3(2*k-1,2*1);
    end;
elseif k==1
    B33(A(k,l),A(k,l))=A3(1,2*1);
end;
if l<m2
    if A3(2*k,2*1+1)==0
        B4(A(k,l),A(k,l+1))=1;
    else
        B44(A(k,l),A(k,l))=A3(2*k,2*1+1);
    end;
elseif l==m2
    B44(A(k,l),A(k,l))=A3(2*k,2*m2+1);
end;
end;
end;
% Utiliza Esparsidade
B1=sparse(B1);
B11=sparse(B11);
B2=sparse(B2);
B22=sparse(B22);
B3=sparse(B3);
B33=sparse(B33);
B4=sparse(B4);
B44=sparse(B44);
C1=sparse(C1);
C1T=sparse(C1T);
C2=sparse(C2);
C2T=sparse(C2T);
C11=sparse(C11);
C22=sparse(C22);
C33=sparse(C33);
C44=sparse(C44);
disp('Preenche matrizes');
% Cria a matriz final de conexao C
C=kron(B1,C2T)+kron(B2,C1T)+kron(B3,C1)+kron(B4,C2)+kron(B11,C44)+kron(B22,C33)+kron(B33,C11)+kron(B44,C22);
% Cria matrizes auxiliares para o calculo da matriz final S
S1=sparse(eye(szx1,szx2));
S2=sparse(1/2*ones(4,4)-eye(4,4));
% Calcula a matriz S final
S=kron(S1,S2);
% Cria a matriz externa auxiliar para o calculo de T
A2=zeros(m1+2,m2+2);
% Preenche a matriz externa auxiliar para calcular T
x1=0;
for k=1:m1+2;
    A2(k,1)=1+x1;
    x1=A2(k,1);
end;
for k=1:m1+2;
    A2(k,m2+2)=1+x1;
    x1=A2(k,m2+2);
end;
for l=2:m2+1
    A2(1,l)=1+x1;
    x1=A2(1,l);
end;
for l=2:m2+1
    A2(m1+2,l)=1+x1;
    x1=A2(m1+2,l);
end;
% Cria matrizes auxiliares para o calculo de T
T1=zeros(2*(m1+m2+4)-4,m1*m2);
T2=zeros(2*(m1+m2+4)-4,m1*m2);
T3=zeros(2*(m1+m2+4)-4,m1*m2);
T4=zeros(2*(m1+m2+4)-4,m1*m2);

```

```

% Preenche as matrizes auxiliares com os dados de A e de A2
for l=2:m2+1
    if A3(1,2*m1-2)==0
        T3(A2(1,1),A(1,1-1))=1;
    else
        T3(A2(1,1),A(1,1-1))=0;
    end;
    if A3(2*m1+1,2*m1-2)==0
        T2(A2(m1+2,1),A(m1,1-1))=1;
    else
        T2(A2(m1+2,1),A(m1,1-1))=0;
    end;
end;
% Fecha as portas cobertas por paredes utilizando os dados de A3
for k=2:m1+1
    if A3(2*k-2,1)==0
        T1(A2(k,1),A(k-1,1))=1;
    else
        T1(A2(k,1),A(k-1,1))=0;
    end;
    if A3(2*k-2,2*m2+1)==0
        T4(A2(k,m2+2),A(k-1,m2))=1;
    else
        T4(A2(k,m2+2),A(k-1,m2))=0;
    end;
end;
% Calcula as matrizes auxiliares C?x
C1x=C1(1,:);
C2x=C2(2,:);
C3x=C1T(3,:);
C4x=C2T(4,:);
% Mais esparsidade para o calculo de T
T1=sparse(T1);
T2=sparse(T2);
T3=sparse(T3);
T4=sparse(T4);
% Calcula a matriz T final
T=kron(T1,C2x)+kron(T2,C1x)+kron(T3,C3x)+kron(T4,C4x);
% Inicio de simulacoes
% Limpa variaveis auxiliares da memoria
clear B1 B2 B3 B4 T1 T2 T3 T4 C1 C2 S1 S2 B11 B22 B33 B44 C1T C2T C1x C2x C3x C4x Cz C11 C22 C33 C44
%clear A A1 A2 A3
% Define numeros de passos de tempo
disp('Inicia calculos')
NT=1500;
% Inicializa os vetores incidentes externos (vei), internos (vii) e refletidos (ver)
ver=zeros(2*(m1+m2+4)-4,1);
vei=zeros(2*(m1+m2+4)-4,1);
vii=ones(4*m1*m2,1);
vax=sparse(zeros(2*(m1+m2+4)-4,4*m1*m2));
% Inicializa o vetor incidente interno para o TLM rapido
vi2=ones(4*m1*m2,1);
% Inicializa vetor de saida vo1
vo1=sparse(zeros(2*(m1+m2+4)-4,NT));
% Inicializa vetor gaussiano
vx1=zeros(1,NT);
% Calcula as matrizes Ax e Bx de y[k]=Ax*w[k] e w[k]=T'*x[k]+Bx*w[k-1]
Ax=sparse(T*S);
Bx=sparse(C*S);
% Inicializa o vetor de excitacao para t=0
vei(6)=exp(-100*((NT/2)/(NT/8))^2);
vx1(1)=vei(6);
% Calcula w[0] e y[0]
vii=T'*vei;
ver=Ax*vii;
U=sparse(eye(4*m1*m2,4*m1*m2));
disp('Inicia TLM lento')
% Calcula vetor de saida para t=0
vo1(:,1)=ver;
% Inicia calculo para t>0
t0=clock;
for k=1:NT-1;
    vei(6)=exp(-100*((k-NT/2)/(NT/8))^2);

```

```

    vx1(k+1)=vei(6);
    vii=T'*vei+Bx*vii;
    ver=Ax*vii;
    vo1(:,k+1)=ver+T*vii;
end;
disp('Tempo das operacoes no TLM lento')
ox1=etime(clock,t0);
disp(ox1);
disp('Inicia matrizes do TLM rapido')
% Inicializa matrizes para o TLM rapido
% Reducao do Modelo - retirada de autovalores
disp('Calculo dos autovalores')
[v1,dx]=eig(full(Bx));
vt=inv(v1);
KT=rank(dx);
q=KT/5;
T1=sparse(vt*T');
A1x=sparse(Ax*v1);
dx=sparse(dx);
disp('Reducao de modelo')
T11=T1(1:q,:);
T21=T1(q+1:4*m1*m2,:);
dx1=dx(1:q,1:q);
A11=A1x(:,1:q);
A12=A1x(:,q+1:4*m1*m2);
AA=T11;
BB=dx1;
CC=A12*T21;
DD=A11;
vio=zeros(q,1);
% Define a razao entre o passo de tempo rapido e o lento
p=2;
% Calcula Matrizes com p pulos de passos de tempo
U=sparse(eye(q));
% Calcula parte das matrizes do TLM rapido
disp('Calculo da matriz A1, B1, C1, D1, E1 e F1')
Ts1=U;
Ts2=0*U;
Bsx=dx1;
for l=1:p-1;
    Ts1=U+dx1*Ts1;
    Ts2=Ts2+l*Bsx;
    Bsx=dx1*Bsx;
end;
A1=sparse((Ts1-Ts2/p)*AA);
B1=sparse(Ts2/p*AA);
C1=sparse(dx1.^p);
D1=sparse(CC);
E1=sparse(DD);
F1=sparse(T*T');
% Limpa as matrizes nao esparsas Ts1, Ts2 e U
clear Ts1 Ts2 U T1 T11 A1x vt v1 dx dx1 T11 T21 A11 A12 AA BB CC DD Bsx
% Numero de passos de tempo do TLM rapido
NT1=NT/p;
% Inicializa vetor de saida vo2
vo2=zeros(2*(m1+m2+4)-4,NT1);
% Inicia algoritimo de calculo do TLM rapido
disp('Inicia TLM rapido')
t0=clock;
for k=1:1:NT1-1;
    xx=vei;
    vei(6)=exp(-100*((p*k-NT/2)/(NT/8))^2);
    vio=A1*vei+B1*xx+C1*vio;
    ver=D1*vei+E1*vio;
    vo2(:,k+1)=real(ver)+F1*vei;
end;
disp('Tempo das operacoes no TLM rapido')
ox2=etime(clock,t0);
disp(ox2);
% Calculo dos vetores de saida
y1=full(vo1(18,:));
y2=full(vo2(18,:));
% Saida grafica do sistema

```

```

disp('Saida Grafica')
figure
% Dominio do tempo
plot(1:NT,y1,1:p*p*NT1,y2,'-+')
xlabel('Passos de Tempo');
ylabel('Amplitude');
title('TLM rapido p=5 - Dominio do Tempo');
% Dominio da Frequencia
h1=fft((y1));
h2=fft((y2));
figure
f1=(0:(NT-1)/10)/(NT-1);
f2=(0:(NT1-1)/10)/(p*(NT1-1));
plot(f1,10*log(abs(h1(1:round(NT/10)))),f2,10*log(abs(p*h2(1:round(NT1/10))),'+-')
xlabel('Passos de Tempo');
ylabel('Magnitude');
title('TLM rapido p=5 - Dominio da Frequencia');
figure
plot(f1,180/pi*angle(h1(1:round(NT/10))),f2,180/pi*angle(p*h2(1:round(NT1/10))),'+-')
xlabel('Passos de Tempo');
ylabel('Fase');
title('TLM rapido p=5 - Dominio da Frequencia');

```

A.3– ARTIGOS PUBLICADOS

A.3.1– Periódicos Indexados

1. C. A. TENÓRIO DE CARVALHO, JR., L.R.A.X. DE MENEZES AND V.H.C. MELO, "Accelerating bidimensional TLM using adjustable time steps", Electronics Letters, Vol. 39 No. 25, 11th December 2003.

A.3.2– Conferência Internacional

1. CARVALHO JÚNIOR, Carlos Alberto Tenório de; MENEZES, Leonardo R A X de; MELO, Victor Hugo C. "High Performance TLM Simulation Using Adjustable Time Steps". International Microwave Symposium IEEE MTT-S, Forth Worth, Texas, June 2004.
2. CARVALHO JÚNIOR, Carlos Alberto Tenório de; MENEZES, Leonardo R A X de. "Extraction of S-Parameter Using the Three-Dimensional Transmission-Line Matrix (TLM) Method". In: International Microwave and Optoelectronics Conference - IMOC. SBMO / IEEE MTT-S. Sociedade Brasileira de Microondas e Optoeletrônica, Foz do Iguaçu, 2003. v.2, p. 967-969.

Artigo Submetido

1. C. A. TENÓRIO DE CARVALHO JR., N. CARVALHO PINHEIRO and L.R.A.X. DE MENEZES. "Complex Envelope Application in Transmission-Line Matrix - John's Super Node Method and Time Steps Estimate in Electromagnetic Simulation". Frontiers in Applied Computational Electromagnetics - FACE 2006, Victoria, Canada, Fev. 2006.

A.3.3– Conferência Nacional

1. CARVALHO JÚNIOR, Carlos Alberto Tenório de; MENEZES, Leonardo R A X de; MELO, Victor Hugo C., R. P. Ribas e André Inácio Reis. "Simulação TLM-2D

de Alto Desempenho Utilizando Técnicas de Aceleração". 11º SBMO - Simpósio Brasileiro de Microondas e Optoeletrônica e a 6º CBMag - Congresso Brasileiro de Eletromagnetismo, MOMAG, São Paulo, 2004.