

Universidade Estadual de Campinas
Faculdade de Engenharia Elétrica e de Computação
Departamento de Microonda e Óptica

Computação Paralela Aplicada a Problemas Eletromagnéticos Utilizando o Método FDTD

Autor: Carlos Henrique da Silva Santos

Orientador: Prof. Dr. Hugo Enrique Hernández Figueroa

Dissertação de Mestrado apresentada à
Faculdade de Engenharia Elétrica e de
Computação da Universidade Estadual de
Campinas como parte dos requisitos necessários
para a obtenção do título de Mestre.

Banca Examinadora

Prof. Dr. Hugo Enrique Hernández Figueroa
Prof. Dr. Antonio Manoel Ferreira Frasson
Prof. Dr. Rui Fragassi Souza

FEEC – UNICAMP
DEL – UFES
FEEC – UNICAMP

Campinas, SP
05 de Agosto de 2005

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

Sa59c Santos, Carlos Henrique da Silva
Computação paralela aplicada a problemas
eletromagnéticos utilizando o método FDTD / Carlos
Henrique da Silva Santos. --Campinas, SP: [s.n.], 2005.

Orientador: Hugo Enrique Hernández Figueroa
Dissertação (Mestrado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Elétrica e de
Computação.

1. Processamento paralelo (Computadores). 2. Diferenças
finitas. 3. Estrutura de domínio. 4. Métodos numéricos. 5.
Maxwell, Equações de. 6. Amostragem por conglomerados.
Software livre. I. Hernández Figueroa, Hugo Enrique. II.
Universidade Estadual de Campinas. Faculdade de
Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: Parallel Computing Applied to Electromagnetic Problems
Using the FDTD Method

Palavras-chave em Inglês: Parallel processing, Finite differences, Domain
structure, Numerical methods, Equations, Maxwell,
Cluster sampling e free software

Área de concentração: Telecomunicações e Telemática.

Titulação: Mestre em Engenharia Elétrica

Banca examinadora: Rui Fragassi Souza e Antonio Manoel Ferreira Frasson.

Data da defesa: 05/08/2005

Resumo

Esse trabalho tem por objetivo desenvolver soluções computacionais de alto desempenho a um baixo custo, seguindo as propostas incentivadoras do Governo Federal para adoção de *software* livre. Essas soluções possibilitam simular, de maneira eficiente, os domínios computacionais de médio e grande porte utilizados no eletromagnetismo computacional. Os bons resultados obtidos nesse trabalho mostram a importância e eficiência da computação massivamente paralela utilizando *cluster Beowulf* para o processamento do método FDTD aplicado em estruturas complexas, porém a um baixo custo financeiro. O desempenho desse sistema ficou comprovado na realização de experimentos para analisar a SAR na cabeça humana e estudar os efeitos de estruturas metamateriais.

Palavras-chaves: Computação Paralela, Eletromagnetismo Computacional, *Cluster Beowulf*, Método das Diferenças Finitas no Domínio do Tempo (FDTD)

Abstract

This work has as objective to develop high performance computational solutions to a low cost, following the stimulated proposals of the Federal Government for adoption of free software. They make possible to simulate, in efficient way, the computational domains of middle and high size useful on the computational electromagnetism. The good results gotten in these work showed the importance and efficiency of the massive parallel computation using the Beowulf cluster for the process the FDTD method applied on complex structures, however to a low financial cost. The performance of this system was proved in the realization of experiments to analyze the SAR on the human head and to study the effects of metamaterial structures.

Key-words: Parallel Computing, Computational Electromagnetism, Beowulf Cluster, Finite Difference Time Domain (FDTD) Method

*Dedico este trabalho aos meus pais
Aurelio e Maria Helena (há 25 anos casados),
ao meu irmão Marco
e às minhas irmãs Ana e Ariane.
À Deus, por tudo.*

Tempo feliz

*Feliz o tempo que passou, passou
Tempo tão cheio de recordações
Tantas canções ele deixou, deixou
Trazendo paz a tantos corações*

*Que sons mais lindos tinha pelo ar
Que alegria de viver
Ah, meu amor, que tristeza me dá
Vendo o dia querendo amanhecer
E ninguém cantar*

*Mas, meu bem
Deixa estar, tempo vai
Tempo vem
E quando um dia esse tempo voltar
Eu nem quero pensar no que vai ser
Até o sol raiar*

Vinícius de Moraes - Poesia completa e prosa: "Cancioneiro"

Agradecimentos

Gostaria de agradecer a todos que de uma forma ou outra contribuíram com esse trabalho.

Agradeço a Deus, pelo dom da feliz vida que ele me presenteou, além de permitir que eu tenha pessoas imensamente importantes ao meu lado;

Não sendo o bastante dedicar, preciso agradecer a minha família pelo amor, paciência e enorme ajuda. Sem eles esse trabalho nunca se realizaria.

Ao Prof. Dr. Hugo E. H. Figueroa e a Profa. Dra. Marli G. F. Hernández, por me orientarem nesse importante degrau de minha carreira profissional.

A Ana Paula, por iluminar a caminhada de minha vida.

Aos amigos Mariana Velcic Maziviero, Gilberto Basseto Júnior e Robson Vieira (meu primo), pela imensa ajuda prestada e principalmente pela companhia.

Aos amigos da cidade de Sorocaba, Ana Paula, Patrícia, André, Edson e Ailton.

Aos sempre amigos Fernando Mandarini Pereira, Fábio Ossamu Oliveira Ogawa e Antonio Cláudio Nobre Crespo.

À Milena, Sartori, Fabiano, Luciano, Socas, Gustavo, Alexandre, Leonardo Ambrósio e demais colegas do DMO.

Aos professores André Franceschi de Angelis, Leonardo Bravo e Kleber Zuza pela ajuda e amizade.

A professora Rosa Maria Machado, por me apresentar e conduzir à Ciência.

A todos os professores que nessa vida já tive, pois seria injustiça não agradecê-los.

A equipe do Laboratório Prof. Dr. Rui Fragassi Souza.

Sumário

Resumo	iii
Abstract	iii
Agradecimentos	vi
Sumário	vii
Lista de Figuras	ix
Lista de Tabelas	xii
Lista de Abreviaturas	xiii
Lista de Símbolos	xiv
Capítulo 1	1
Introdução e Objetivos	1
1.1 Objetivos	1
1.2 Objetivos e Justificativas	2
1.3 Organização do trabalho	3
Capítulo 2	4
Computação de Alto Desempenho	4
2.1 Introdução	4
2.2 Computação Distribuída	5
2.3 Computação Paralela	7
2.4 Cluster Beowulf	9
2.4.1 Segurança do Cluster	11
Capítulo 3	13
Equacionamento das Diferenças Finitas no Domínio do Tempo para Computação Paralela	13
3.1 Introdução	13
3.2 Equações de Maxwell	14
3.3 Diferenças Finitas no Domínio do Tempo (FDTD)	16
3.4 Camadas Perfeitamente Casadas – Perfectly Matched Layer (PML)	27
3.5 FDTD em Paralelo	31

Capítulo 4	35
Desenvolvimento e Discussão dos Resultados	35
4.1 Introdução	35
4.2 Estruturas de Desenvolvimento e Testes	36
4.2.1 Iniciando as Implementações e Analisando os Resultados	36
4.2.2 Desenvolvimento e Resultados de Desempenho do FDTD 2D e 3D em Paralelo	39
4.2.3 FDTD Paralelo para Análise da SAR na Cabeça Humana	52
4.2.4 FDTD Aplicado aos Estudos de Metamateriais	58
Capítulo 5	65
Conclusões e Trabalhos Futuros	65
5.1 Conclusões	65
5.2 Trabalhos Futuros	66
Referências Bibliográficas	68
Anexos	73
Anexo A	73
Instalando e Configurando um Cluster Beowulf	73
Anexo B	77
Instalando, Configurando e Utilizando a LAM-MPI	77
Anexo C	80
Leis de Amdahl	80

Lista de Figuras

Fig. 2.1 – Exemplo de arquitetura distribuída.	6
Fig. 2.2. – Pacotes de mensagens trafegando na rede [14].	8
Fig. 2.3 – Esquema de DSI apresentado por Pourzandi <i>et al.</i> [21].	11
Fig. 3.1 – Esquema de acoplamento dos campos elétricos e magnéticos.	17
Fig. 3.2 – Esquema leapfrog time-step.	18
Fig. 3.3 – Quatro Células de Yee interligadas.	22
Fig. 3.4 – Domínio computacional Ω cartesiano e bidimensional, apresentado por Moore [41].	27
Fig. 3.5 – Mini-câmara anecóica construída por Sartori para seus experimentos com metamateriais [43]	28
Fig. 3.6 – Exemplo de uso da PML num guia de onda retangular.	30
Fig. 3.7 – Esquema de paralelização do domínio computacional tridimensional do FDTD em três processos.	31
Fig. 3.8 – Troca de pacotes entre dois processos.	32
Fig. 3.9 – Troca de mensagens entre processos do FDTD paralelo no modo TE_z	33
Fig. 4.1 – Primeira estrutura de <i>Cluster</i> da LE-45.	37
Fig. 4.2 – Testes de desempenho através da multiplicação vetor x matriz.	39
Fig. 4.3 – Gráfico com o ganho computacional obtido no primeiro cluster do AEL.	40
Fig. 4.4 – Esquema do primeiro <i>Cluster</i> montado no AEL (CLUSTER-AEL-1).	41
Fig. 4.5 – Resultados da cavidade ressonante em diferentes intervalos de tempo (a) 100 passos, (b) 400, (c) 700 e (d) 800 passos de tempo.	41
Fig. 4.6 – Infra-estrutura de segurança adotado nos <i>clusters Beowulf</i> seguindo o modelo de Pourzandi <i>et al.</i> [21]	42
Fig. 4.7 – Gráfico do desempenho dos recursos do AEL1, com 100 passos de tempo.	43

Fig. 4.8 – Gráfico de desempenho dos recursos do AEL3, com 100 passos de tempo.	43
Fig. 4.9 – Configuração atual do <i>cluster Beowulf</i> do AEL, (CLUSTER-AEL-2).	44
Fig. 4.10 – Tempo despendido na simulação 2D variando os passos de tempo.	44
Fig. 4.11 – Comparativo de desempenho entre as estruturas de <i>cluster</i> .	45
Fig. 4.12 – Comparando a rapidez para solucionar o FDTD 2D em Paralelo.	46
Fig. 4.13 – Eficiência computacional do processamento paralelo.	46
Fig. 4.14 – Cálculo da máxima eficiência pela Lei de Amdahl (<i>speedup</i>).	46
Fig. 4.15 – <i>Speedup</i> obtido na cavidade ressonante 3D.	47
Fig. 4.16 – Eficiência da paralelização do FDTD 3D (100x100x100).	48
Fig. 4.17 – Eficiência do paralelismo do FDTD 2D melhorado.	49
Fig. 4.18 – Análise do desempenho do paralelismo do FDTD 2D pela de Amdahl.	50
Fig. 4.19 – Tempo gasto nas simulações de médio e grande porte da cavidade ressonante 2D.	51
Fig. 4.20 – <i>Speedup</i> da cavidade ressonante 2D.	51
Fig. 4.21 – Cortes transversais (12, 15, 25, 35 e 45) com os tecidos mais importantes.	53
Fig. 4.22 – Corte 1 dividido entre 3 processos	54
Fig. 4.23 – Tempo gasto para as simulações da SAR na cabeça humana.	54
Fig. 4.24 – Fonte e campo E_y no trigésimo passo de tempo da Simulação.	55
Fig. 4.25 – Resultado da SAR na cabeça humana após 1000 passos de tempo.	56
Fig. 4.26 – Tempo despendido na simulação a SAR 2D da cabeça humana.	57
Fig. 4.27 – Eficiência da paralelização da simulação 2D da SAR na cabeça humana.	57
Fig. 4.28 – Eficiência da Paralelização da SAR 2D (400x400), variando os	57

passos de tempo.

Fig. 4.29 – Lei de Snell com uma reflexão negativa.	59
Fig. 4.30 – Desempenho da Simulação de Estruturas Metamateriais variando as dimensões.	60
Fig. 4.31 – Estrutura metamaterial utilizada na simulação do campo 300x810.	60
Fig. 4.32 – Análise de desempenho da paralelização do FDTD para estruturas metamateriais	61
Fig. 4.33 – Estrutura metamaterial em paralelo para análises dos efeitos físicos	61
Fig. 4.34 - Tempo despendido para simular a estrutura metamaterial em vários processos	62
Fig. 4.35 – <i>Speedup</i> da simulação da estrutura metamaterial	62
Fig. 4.36 – Banda Proibida da simulação da estrutura metamaterial (Db)	63
Fig. 4.37 – Anel base da estrutura metamaterial	64
Fig. A1. – Representação de um cluster	74

Lista de Tabelas

Tabela 3.1 – Descrição dos símbolos com sua respectiva unidade de medida	16
Tabela 4.1 – Valores n para dimensionamento das simulações	37
Tabela 4.2 – Tempo despendido, em segundos, para simulações de cavidades ressonantes 2D.	41
Tabela 4.3 – Índice de propriedades dos materiais da cabeça humana	54

Lista de Abreviaturas

ABC	Absorption Boundary Condition
DMO	Departamento de Microonda e Óptica
DSI	Distributed Security Infrastructure
FDDI	Fiber Distributed Data Interface
FDM	Finite Difference Method
FDTD	Finite Difference Time-Domain
FEM	Finite Element Method
GPL	Gnu Public License
HA	High Availability
HP	High Performance
IBM	International Business Machine
IP	Internet Protocol
IEEE	Institute of Electrical and Electronics Engineers
LAM-MPI	Local Area Multicomputer - MPI
LAN	Local Area Network
MCT	Ministério da Ciências e Tecnologia
MoM	Moment Method
MPI	Message Passing Interface
MPP	Massively Parallel Process
PML	Perfectly Matched Layer
PVM	Parallel Virtual Machine
QoS	Quality of Service
RBC	Radiation Boundary Condition
RFC	Request for Comments
SAR	Specific Absorption Rate
TCP	Transfer Control Protocol
Wi-Fi	Wireless Fidelity

Lista de Símbolos

A_m	Valor de Amdahl
\vec{B}	Densidade de Fluxo Magnético
\vec{D}	Densidade de Fluxo Elétrico
\vec{E}	Campo Elétrico
\vec{H}	Campo Magnético
\vec{J}	Densidade de Corrente Elétrica
m	Massa
\vec{M}	Densidade de Corrente Magnética
ρ	Número de Processos
Sp	Speedup
t	Tempo
v	Volume
W	Taxa de Aumento da Energia Absorvida por um Elemento
ϵ_r	Permissividade Relativa
ϵ_0	Permissividade do Espaço Livre
μ_r	Permeabilidade Relativa
μ_0	Permeabilidade do Espaço Livre
σ	Condutividade Elétrica
σ^*	Condutividade Magnética
Ω	Domínio Computacional

Capítulo 1

Introdução e Objetivos

1.1 Objetivos

O termo “Sociedade da Informação” passou a ser utilizado nos últimos anos substituindo a “Sociedade pós-industrial”. Essa mudança é uma alusão ao novo modelo técnico-econômico que considera a grande quantidade de informações úteis que são proporcionadas pelos avanços tecnológicos na microeletrônica e telecomunicações [1]. No entanto, esse avanço não é apenas restrito a essas áreas descritas por Wethein [1]. O eletromagnetismo, que é o foco deste trabalho, também evoluiu bastante desde o século XIX, quando James Clerk Maxwell afirmou que a eletricidade e o magnetismo se combinam para formar a luz [2]. A partir da concepção das Equações de Maxwell muitas aplicações surgiram com diferentes soluções no eletromagnetismo, ressaltando que a complexidade também tem variado.

A quantidade de recursos computacionais necessários às simulações dos métodos numéricos aplicados ao eletromagnetismo, no geral, é grande. Por esse motivo a exploração de técnicas e tecnologias aplicadas no eletromagnetismo computacional, em geral, visa aperfeiçoar as soluções que demandam grande poder de processamento, reduzindo o tempo despendido pelas tarefas. Para exemplificar, em seu trabalho Claudio [3] apresenta métodos numéricos eficientes para manipulações matriciais, contribuindo para solucionar campos vetoriais do eletromagnetismo computacional.

Quanto aos recursos computacionais para problemas de grande porte, atualmente destaca-se o uso do processamento massivamente paralelo através do *cluster Beowulf*, que é um conjunto de computadores interligados por uma rede, com o objetivo de solucionar problemas de grande porte utilizando *softwares* gratuitos para a redução dos gastos financeiros. Por esse

motivo, vem sendo bastante aplicado em problemas de grande porte devido ao baixo custo, contornando a demanda por supercomputadores.

A redução nas despesas é devido ao *Beowulf* ter como fundamento o uso de *software* gratuito e computadores de comércio fácil, viabilizando a construção de um sistema escalonável e de fácil manutenção. Assim, tem-se um mecanismo que possibilita a solução de problemas inviáveis ou incapazes de serem resolvidos em um único computador, porém sem um supercomputador.

A proposta desse trabalho é analisar a utilização de *clusters Beowulf*, com implementações de troca de mensagens, que viabilizem a solução de problemas numéricos de médio e grande porte. Os *clusters* são utilizados para possibilitar que os métodos numéricos do eletromagnetismo computacional como, por exemplo, o método dos elementos finitos, das diferenças finitas e o método dos momentos, sejam mais facilmente solucionados. Nesse trabalho optou-se pelo método das diferenças finitas no domínio do tempo, devido à facilidade de paralelização e inúmeras aplicações possíveis, como em estudos da SAR na cabeça humana e estruturas metamateriais.

1.2 Objetivos e Justificativas

O objetivo deste trabalho é aproveitar o parque computacional disponível nos laboratórios do DMO para construir *clusters Beowulf* que possibilitem realizar simulações eletromagnéticas de médio e grande porte utilizando como ferramenta numérica o método das diferenças finitas no domínio do tempo (FDTD). Essa estrutura soluciona problemas complexos através do processamento paralelo, porém reduzindo os custos financeiros. Outra importante questão que valoriza o *cluster Beowulf* é a quantidade de documentação de fácil acesso na Internet [4], que agiliza e facilita o trabalho.

Com esses recursos, são feitas propostas de soluções computacionais alternativas ao grupo, utilizando *software* livre para resolver problemas de grande porte, proporcionando maior flexibilidade aos seus trabalhos e seguindo os atuais incentivos do Governo Federal, que através do Ministério de Ciências e Tecnologia (MCT), considera as plataformas abertas e livres, como possíveis meios de valorização e inovação das tecnologias do setor de *software*. Isso

vem ocorrendo desde um balanço realizado pelo MCT no ano de 2003, que também tem considerado que a adoção de *software* livre para a Administração Pública pode resultar em significativa redução nos custos de *hardware*, *software* e serviços, além de proporcionar maior flexibilidade para a definição de estratégias na área de tecnologia da informação [5].

Dessa forma, tem-se um trabalho que se preocupa com a redução dos gastos financeiros para o provimento de recursos computacionais para solucionar problemas de médio e grande porte, seguindo os incentivos do Governo Federal. Sendo assim, pode-se definir que a meta deste trabalho é propor o aproveitamento do parque computacional do departamento e o uso de *software* livre em simulações de campos eletromagnéticas de grande porte a um baixo custo financeiro, utilizando como ferramenta numérica o método FDTD.

Os resultados obtidos por esse trabalho são bons. As implementações dos *clusters Beowulf*, utilizando o parque computacional do DMO, possibilitaram simular campos eletromagnéticos de médio e grande porte, através do método FDTD, sem acarretar qualquer gasto financeiro extra ao departamento, os quais seriam investidos na aquisição de outros equipamentos e de aplicativos. O desempenho do paralelismo também foi bom, reduzindo bastante o tempo de espera da solução.

1.3 Organização do trabalho

No Capítulo 2 são apresentados conceitos sobre computação distribuída e paralela, itens necessários ao entendimento da aplicação do *cluster Beowulf*, abordando sua definição, surgimento e os possíveis tipos de aplicações. O equacionamento do método das diferenças finitas no domínio do tempo (FDTD) é mostrado no Capítulo 3, empregando a PML e, por fim, é mostrado como os algoritmos FDTD foram paralelizados. As implementações e os resultados são discutidos no Capítulo 4, que também traz as aplicações do método em simulações de SAR na cabeça humana e nos estudos de estruturas metamateriais. No quinto e último capítulo, o conteúdo trata das conclusões obtidas a partir dos resultados, além de trazer as possibilidades de trabalhos futuros.

Capítulo 2

Computação de Alto Desempenho

2.1 Introdução

A história dos dois últimos séculos é considerada por Castells [6] como sendo um dos raros intervalos da história no qual segue o “gradualismo” descrito pelo paleontólogo Stephen J. Gould. Ele afirma que a história da vida é uma série de situações estáveis, pontuadas por raros intervalos de eventos importantes que ocorrem com grande rapidez e ajudam a estabelecer a próxima era estável. Essa afirmação baseia-se, principalmente, no atual processo de transformação e exponencial expansão tecnológica, decorrente das capacidades de interação entre os diversos campos tecnológicos através de uma linguagem comum que possibilita que a informação seja gerada, armazenada, recuperada, processada e transmitida.

A expansiva velocidade do desenvolvimento tecnológico contribui com a geração de necessidades, que muitas vezes, não são satisfeitas e acabam demandando novos recursos, ocasionando um ciclo interminável. Esse ciclo é notado na computação quando colocados numa balança a evolução de *hardware* e *software*. Para ser mais específico, a relação entre o desenvolvimento de novas tecnologias dos processadores e memórias, e as novas técnicas dos aplicativos.

Humphries [7] afirma que há uma explosão sobre a evolução de métodos numéricos, fato que vem ocorrendo na última década. Com isso, há a exigência de computadores que atendam essa demanda. Porém, a evolução física dos componentes, atualmente, não acompanha a evolução dos *softwares*, principalmente os utilizados em simulações numéricas aplicadas nas ciências e engenharias [8], em especial no eletromagnetismo computacional. Por esse

motivo, são necessárias soluções alternativas que amenizem ou que resolvam definitivamente essa questão, que pode implicar na redução do ritmo de evolução científica e tecnológica.

Uma solução alternativa e viável é o processamento massivamente em paralelo (MPP), que divide um problema de grande porte e o distribui entre os processos de um sistema. Porém, deve-se considerar que os custos podem ser muito elevados como, por exemplo, a aquisição de um supercomputador que é inviável para a grande maioria das instituições públicas e privadas.

A opção pelo *cluster Beowulf* é uma boa alternativa. Ele é um conjunto de computadores, podendo ser estações trabalhos, interligados através de uma rede para solucionar um determinado problema que requer alto desempenho. A essência de funcionamento desse tipo de *cluster* é baseada no uso de *softwares* livres, que reduzem ainda mais os custos.

2.2 Computação Distribuída

Existem diversas definições sobre os sistemas de processamento distribuído, porém a que melhor se adapta a esse trabalho é a apresentada por Veríssimo e Rodrigues [9], que primeiramente define as redes de computadores como sendo uma infra-estrutura de interligação de um conjunto de computadores com diversas topologias e meios de comunicação, que possuem os mesmos protocolos para transferência de dados. Eles especificam que os sistemas distribuídos são compostos por diversos computadores que se comunicam através de redes e que hospedam os processos que são executados em um conjunto comum de protocolos distribuídos, que administram a execução das atividades distribuídas. Na Fig. 2.1 é apresentado um sistema distribuído em que as estações das diferentes redes internas (*intranets*) são utilizadas para o processamentos paralelo.

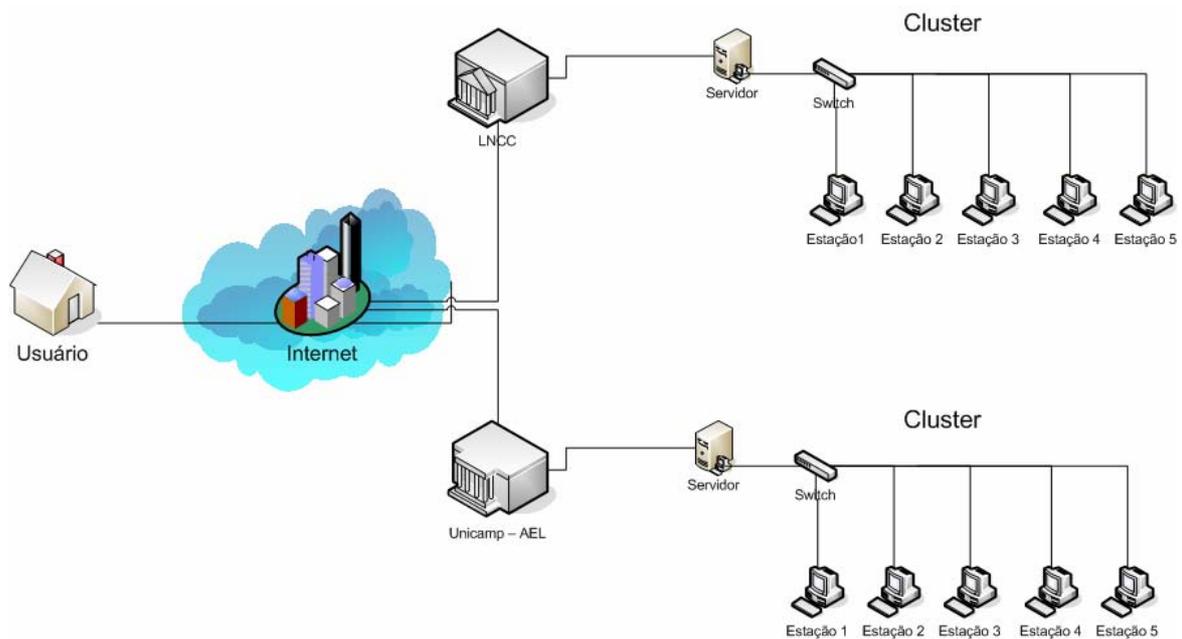


Fig. 2.1 – Exemplo de arquitetura distribuída

Os sistemas distribuídos, na grande maioria, são complexos de serem implantados e difíceis de serem gerenciados. Por esses motivos, é muito importante que qualquer projeto dessa natureza seja precedido dos levantamentos de aspectos tecnológicos e estruturais que estarão presentes no sistema [10]. Tais aspectos são apresentados a seguir:

- planta física do projeto;
- informações da rede;
- tipos de comunicações entre processos;
- tipos de transações;
- sistemas operacionais.

A planta física do projeto implica na estrutura física das redes de computadores, que por sua vez, influencia o padrão de rede a ser utilizado. É fundamental que seja de conhecimento do arquiteto de redes os padrões como, por exemplo, *Ethernet*, ATM, FDDI, Wi-Fi, pois permite aplicar diferentes soluções que melhor se adaptem a cada problema, seguindo normas internacionais, como a norma IEEE 802 [11], que descreve a família de padrões publicados pela IEEE para redes de áreas locais (*Local Area Networks – LANs*) e para as redes metropolitanas (*Metropolitan Area Network – MANs*).

Para elucidar o uso dessa norma, um dos bons e mais utilizados modelos de referência para padrões de protocolos, contidos na norma IEEE 802, é o padrão IEEE 802.3 que especifica a *Ethernet*.

Existem outras duas informações importantes que devem ser de conhecimento do responsável pelo projeto. A primeira está relacionada aos sistemas operacionais que possivelmente serão utilizados na arquitetura distribuída como, por exemplo, Microsoft Windows (2000, NT, XP), Unix (Solaris, HP-UX, IRIS), Linux (Fedora, Conectiva, Debian Slackware), Apple, entre outros. A segunda informação é conhecer os aplicativos que serão utilizados, pois em muitos casos é possível que os serviços do sistema operacional sejam otimizados para eles.

Após esse levantamento, os arquitetos, ou engenheiros, de sistemas de computação podem propor algumas soluções computacionais melhor adaptadas à distribuição das tarefas, sendo que essas podem ser executadas através de processos ou *threads*, que segundo Tanenbaum e Steen [12], podem ser definidos como:

- *Processo* é um programa em execução, isto é, um programa que está sendo executado num dado momento no processador virtual do sistema operacional devidamente referenciado como um.
- *Threads* é bem parecido com o processo em seu funcionamento no processador virtual, porém existe um alto grau de transparência da concorrência, pois apresenta mais identificadores, o que pode prejudicar o desempenho.

2.3 Computação Paralela

A computação paralela, em geral, está subentendida no conceito de computação distribuída, porém, dependendo da aplicação o paralelismo não ocorre. Por isso, a importância de se definir a computação paralela, sendo que ela ocorre quando, num dado instante, existe mais de um processo trabalhando na resolução de um mesmo problema [13].

O uso de supercomputadores é uma outra solução eficiente para o processamento paralelo através do “balanceamento de cargas” ou mecanismos de troca de mensagens. O balanceamento de cargas é um mecanismo que

divide automaticamente as tarefas entre os processadores, independente se elas são seqüenciais ou paralelas. O outro mecanismo é a troca de mensagens, em que um processo se comunica com os outros para trocar dados, sendo considerada uma “solução explícita” devido à forma como os processos se comunicam. Na Fig. 2.2, é apresentado um esquema de troca de mensagens entre os processos 0 e 2, no qual o pacote transmitido aparece trafegando pela rede.

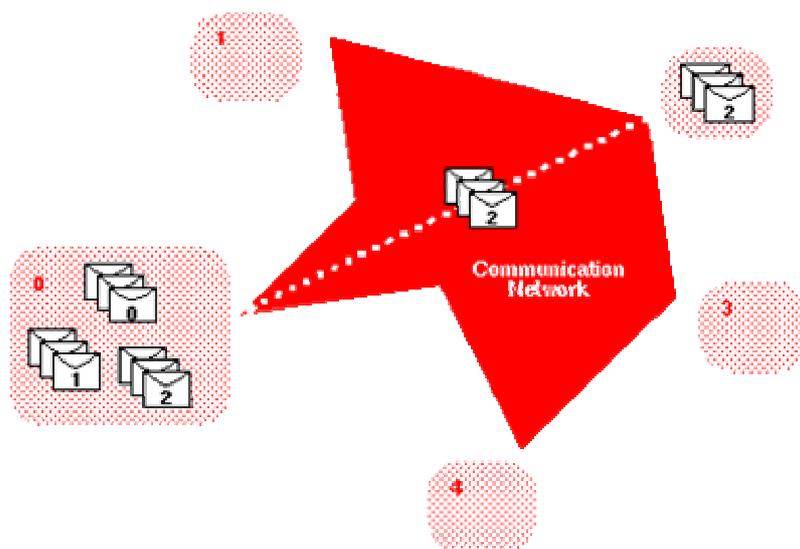


Fig. 2.2. – Pacotes de mensagens trafegando na rede [14]

No desenvolvimento de aplicativos paralelos de troca de mensagens, a implementação pode seguir as características da estrutura em que os dispositivos físicos são alocados. Para exemplificar, nas arquiteturas de multiprocessamento o sistema de memória pode ser compartilhado, distribuído, ou uma combinação desses [15], variando as possibilidades de sua construção. Essas arquiteturas são encontradas nos supercomputadores que são bastante caros e estão além dos recursos disponíveis por grande parte das instituições públicas e privadas. Para contornar esse problema, as instituições têm adotado o *cluster Beowulf*, que possibilita o processamento massivo em paralelo a custos bem menores e com capacidades de processamento semelhantes aos dos supercomputadores. No Anexo A, é mostrado uma implementação passo-a-passo simples para o *cluster Beowulf*.

2.4 Cluster Beowulf

O *cluster* de computadores é definido por Simões *et al.* [16] como sendo um conjunto de computadores interligados em rede e configurados para trabalharem em paralelo para solução de um problema. Os serviços providos pelo *cluster* de computadores são divididos por Pitanga [17], basicamente, em duas categorias: os de alta disponibilidade (HA) e alto desempenho (HP).

A finalidade dos serviços de alta disponibilidade é oferecer ininterrupto acesso a tarefas computacionais confiáveis a determinados grupos de usuários como, por exemplo, os recursos disponibilizados através da Internet pelos bancos, onde seus usuários podem realizar transações a qualquer hora do dia, com restrições de acesso através de sistemas de segurança. Nesse caso, várias máquinas podem ser acessadas praticamente ao mesmo tempo, retornando o resultado das referidas consultas aos usuários, garantindo a integridade e disponibilidade dos dados. Um exemplo é o sistema bancário para prestação de serviços via Internet, devido à sua disponibilidade de uso 24 horas por dia, 7 dias da semana, tarefa assegurada por um conjunto de servidores e equipamentos que realizam essa tarefa de alta disponibilidade.

Outro tipo de serviço especificado por Pitanga [17], que é o foco deste trabalho, está no uso de *cluster* de computadores para resolver problemas de computação de alto desempenho, ou seja, que demandam grande capacidade de armazenamento e/ou processamento e que não são possíveis de serem realizados em estações de trabalhos comuns. A generalização do termo processamento, neste caso, estende-se à resolução de problemas complexos, que utilizam muitos recursos computacionais, como processadores e memórias (temporárias ou permanentes).

Os trabalhos nessa linha começaram a ser apresentados no *Center of Excellence in Space Data and Information Sciences* (Cesdis), em 1994, que é uma divisão da *University Space Research Association* (USRA) [18], devido às suas necessidades de processamento de grande porte e gerenciável a um baixo custo [19]. Esses fatores culminaram na definição oficial do *cluster Beowulf*, que tem esse nome por fazer referência ao primeiro livro impresso da literatura inglesa [20]. Por definição, esse é um tipo de *cluster* escalonável de produtos de *hardware*, para um sistema de rede privada e com infra-estrutura

de *software* de código-aberto e gratuito como, por exemplo, o sistema operacional Linux, utilizando compiladores das linguagens C, C++ e Fortran. Em complemento, também é importante destacar que uma das principais preocupações das pessoas envolvidas com *cluster Beowulf* é torná-lo comercialmente disponível, com vasta documentação e gerenciável por mão-de-obra qualificada [17]. Analisando e seguindo todos esses itens, essa infraestrutura proporciona uma boa economia nos custos aquisitivos de processamento para soluções de alto desempenho, amenizando, ou solucionando, a necessidade de recursos computacionais mais potentes.

O *cluster Beowulf* também acabou contribuindo em outras áreas como, por exemplo, algumas melhorias no núcleo (*kernel*) padrão do Linux e no esquema de pilhas do protocolo TCP/IP. Devido a todas essas vantagens, tal tecnologia vem sendo facilmente disseminada e praticada, principalmente, pelas instituições de ensino e pesquisa [17].

Em relação ao funcionamento, o *cluster* apresenta um simples princípio de uso tecnológico, em que se tem uma tarefa complexa e exigente de grande poder computacional, ela é decomposta e distribuída em partes independentes entre os vários processos que fazem parte do *cluster*, ou seja, tornar um problema grande e complexo passível de ser resolvido por muitos computadores, agilizando o trabalho. Essa técnica é difícil de ser idealizada computacionalmente, pois os algoritmos paralelos são mais difíceis de serem implementados [15].

Para a realização do processamento paralelo Simões *et al.* [16], cita a importância de destacar alguns requisitos do *cluster* de computadores para que ele possa atender à demanda requisitória de uso. É interessante que esses itens sejam definidos no pré-projeto, sendo:

- Segurança da infra-estrutura (rede, computadores, refrigeração e instalação elétrica);
- Qualidade no serviço (QoS);
- Normas de uso;

A qualidade dos serviços e as normas de uso do *cluster* variam bastante, estando geralmente relacionados com a quantidade de usuários e as tarefas que eles realizam. Já a segurança da infra-estrutura é importante para todos os

projetos, principalmente a segurança da rede que interconecta os computadores.

2.4.1 Segurança do Cluster

De maneira geral, os dados que são processados em um sistema de computadores, como o *cluster*, são informações particulares, publicáveis ou não. Dessa forma, é importante que existam mecanismos de segurança sobre esses dados, que atendam as necessidades e expectativas dos usuários.

Pourzandi *et al.* [21] apresentam alguns elementos para uma infraestrutura distribuída segura (DSI) como, por exemplo, nível dos processos individualizado, políticas de segurança dinâmica, chaves de gerenciamento transparente utilizando criptografia e acesso controlado. Na Fig. 2.3 é possível notar como este esquema funciona de acordo com sua hierarquia de organização.

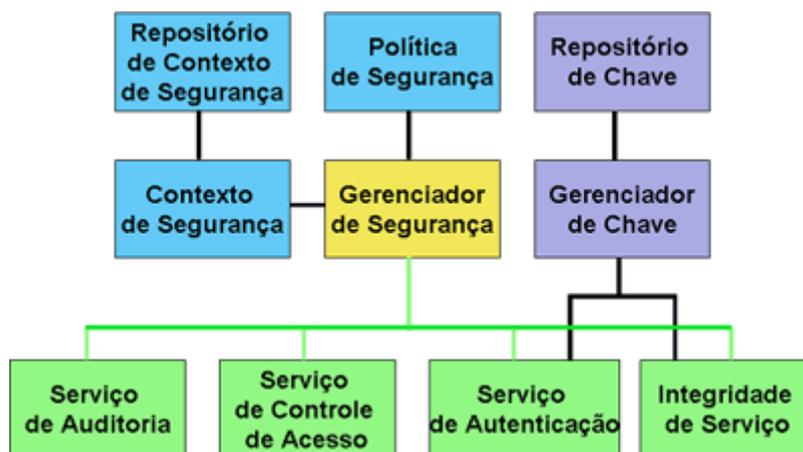


Fig. 2.3 – Esquema de DSI apresentado por Pourzandi *et al.*[21].

Uma forma de valorizar a estrutura é determinar os diferentes níveis de segurança para os acessos como, por exemplo, num modelo médio os usuários externos a rede tem determinada liberdade de acesso ao equipamento, sem muitas filtragens no meio do caminho. Esse tipo de configuração é interessante quando não existem muitas pessoas utilizando o sistema ou quando possui pouquíssimos usuários de acesso. Também existe o modelo de acesso restrito, que é adotado pelas companhias e agência governamentais que utilizam *firewalls* com regras bastante avançadas e que bloqueiam grande parte das tarefas de entrada e saída de dados da rede [22].

Algumas técnicas contribuem com o equilíbrio dos modelos de *firewall*. Para exemplificar Gorsuch [23] afirma que mascarar o IP, seguindo a RFC 1918, é uma interessante implementação de segurança, pois permite que o *cluster* permaneça “escondido” dentro da rede.

Existem muitas outras possibilidades de configuração de sistema de segurança de uma rede, variando de acordo com o nível esperado e o custo financeiro. Por esse motivo, é muito importante esclarecer as prioridades para que seja realizada uma instalação de acordo com as necessidades dos usuários, fazendo um balanceamento entre desempenho, usabilidade e segurança. Essa é uma relação relatada por Yucik *et al.* [24] como sendo o estado da arte para o planejamento de serviços de segurança.

Capítulo 3

Equacionamento das Diferenças Finitas no Domínio do Tempo para Computação Paralela

3.1 Introdução

O eletromagnetismo iniciou-se no Século XIX com um rápido desenvolvimento no campo da eletricidade e do magnetismo, dentre os quais são citados os trabalhos pioneiros de Oersted (1771-1851), Faraday (1791-1867) e Ampère (1775-1836) [25]. Porém, foi James Clerk Maxwell (1831-1879) quem publicou pela primeira vez a teoria eletromagnética como é praticada no presente. Ele também predisse as bases das ondas eletromagnéticas [26]. No entanto, seus trabalhos só foram plenamente aceitos pela comunidade científica após o professor de física Heinrich Rudol Hertz (1857-1894) gerar, com sucesso as ondas de rádio, comprovando as teorias de Maxwell.

Após essa comprovação, atribui-se a Maxwell a generalização das várias formas das leis da eletricidade e do magnetismo, que ficaram conhecidas como equações de Maxwell. Ele também relacionou suas bases teóricas para o estabelecimento da natureza eletromagnética da luz [27].

Todos esses fundamentos contribuíram bastante com a rápida evolução tecnológica e científica, sendo que a essência dos problemas ligados ao eletromagnetismo está cada vez mais complexa, devido às novas e variadas aplicações. Alguns desses problemas são impossíveis de serem resolvidos analiticamente, por esse motivo são empregadas técnicas numéricas solucionadas com o auxílio dos computadores, realizando as simulações necessárias aos estudos e comprovações teóricas.

O método das diferenças finitas no domínio do tempo (FDTD), idealizado por Kane S. Yee [28] no ano de 1966, é um dos mais utilizados e apresenta satisfatória precisão. O FDTD permite que as Equações de Maxwell sejam

discretizadas no tempo e no espaço, com certa aproximação, possibilitando simular efeitos e dispositivos eletromagnéticos que variam com o passar do tempo. Dessa forma, tem-se a importância do eletromagnetismo computacional para a modelagem e simulação de propriedades eletromagnéticas. Para elucidar as importantes aplicações é interessante iniciar-se nos esforços empregados em trabalhos que analisam os efeitos das taxas relativas de absorção das ondas eletromagnéticas no corpo humano [29],[30],[31], aplicações em telecomunicações como, antenas[32],[33], radares [34], satélites [35] e fibras óptica. Existem inúmeras outras áreas de concentração que também podem variar a complexidade da modelagem e da simulação.

O FDTD é um método que pode ser aplicado nessas tarefas, por ser eficiente e retornar resultados satisfatórios, porém em alguns desses casos os computadores do tipo estação de trabalho não são capazes de solucionar os problemas, havendo a necessidade do uso de supercomputadores, que, devido ao custo, são substituídos por tecnologias e técnicas alternativas, como a computação massivamente paralela utilizando *cluster* de computadores, em especial os do tipo *Beowulf*, já descrito.

Nesse capítulo é apresentado o equacionamento necessário à discretização das Equações de Maxwell (Seção 3.2). Na Seção 3.3 é mostrado o equacionamento do método das diferenças finitas no domínio do tempo (FDTD), com seu algoritmo. A seguir, na Seção 3.4, apresenta-se o recurso das camadas perfeitamente casadas (PML), desenvolvida por Bérenger. A última seção (Seção 3.5) fica incumbida de elucidar o paralelismo no método FDTD, para as simulações eletromagnéticas.

3.2 Equações de Maxwell

As representações do mundo real ocorrem nos computadores através de simulações que são criadas, na grande maioria, através de modelos matemáticos e, no eletromagnetismo computacional, não poderia ser diferente. Existem muitas ferramentas de base numérica que podem simular condições reais, seguindo as leis que governam o eletromagnetismo, isto é, pelas Equações de Maxwell [36] que são apresentadas a seguir.

$$\frac{\partial \vec{B}}{\partial t} = -\nabla \times \vec{E} - \vec{M}, \quad (3.1)$$

$$\frac{\partial \vec{D}}{\partial t} = \nabla \times \vec{H} - \vec{J}, \quad (3.2)$$

$$\nabla \cdot \vec{D} = 0, \quad (3.3)$$

$$\nabla \cdot \vec{B} = 0, \quad (3.4)$$

sendo \vec{D} , \vec{B} , \vec{J} e \vec{M} :

$$\vec{D} = \epsilon_r \epsilon_0 \vec{E}, \quad (3.5)$$

$$\vec{B} = \mu_r \mu_0 \vec{H}, \quad (3.6)$$

$$\vec{J} = \sigma \vec{E}, \quad (3.7)$$

$$\vec{M} = \sigma^* \vec{H}, \quad (3.8)$$

onde, $\epsilon_r, \epsilon_0, \mu_r, \mu_0, \sigma, \sigma^*$ são parâmetros conhecidos dos materiais, \vec{J} e \vec{M} são as densidades das correntes elétrica e magnéticas respectivamente. A permissividade do espaço livre é denotada por ϵ_0 , a permissividade relativa; ϵ_r , a permeabilidade no espaço livre é μ_0 e μ_r é a permeabilidade relativa, além disso, o campo elétrico é representado por \vec{E} e o magnético por \vec{H} . Na Tabela 3.1, a seguir, são mostrados os significados de cada símbolo das equações de campos eletromagnéticos, com suas respectivas unidades de medições, especificadas no padrão MKS.

Tabela 3.1 – Descrição dos símbolos com sua respectiva unidade de medição

Símbolo	Significado (unidade de medição)
\vec{E}	Campo elétrico (volt/metro)
\vec{D}	Densidade de fluxo elétrico (coulomb/metro ²)
\vec{H}	Campo magnético (ampere/metro)
\vec{B}	Densidade de fluxo magnético (weber/metro ²)
\vec{J}	Densidade de corrente elétrica (ampere/metro ²)
\vec{M}	Densidade de corrente magnética equivalente (volt/metro ²)
σ	Condutividade elétrica (siemens/metro)
σ^*	Condutividade magnética (ohm/metro)

No eletromagnetismo computacional, várias técnicas numéricas podem ser empregadas nas simulações das diversas áreas das engenharias e ciências. Entre os muitos possíveis métodos de serem escolhidos, as principais resoluções são realizadas através das equações diferenciais parciais com o Método das Diferenças Finitas (FDM) e Métodos dos Elementos Finitos (FEM), que resultam em matrizes esparsas. Também podem ser solucionadas através das equações integrais que são convertidas em equações matriciais cheias utilizando o Método dos Momentos (MOM) [37]. Neste trabalho, que visa o provimento de um mecanismo computacional para solucionar problemas de alto desempenho de maneira eficiente e seguro, optou-se pelo método das diferenças finitas no domínio do tempo, devido à sua facilidade de paralelização e à vasta possibilidade de aplicação no eletromagnetismo computacional.

3.3 Diferenças Finitas no Domínio do Tempo (FDTD)

As diferenças finitas no domínio do tempo possibilitam a discretização das Equações de Maxwell, através de um sistema de seis equações diferenciais parciais acopladas para representar as interações das ondas eletromagnéticas com objetos em três dimensões. Seguindo a definição apresentada por Taflove e Hagness [37], os campos elétricos são descritos pelas equações:

$$\frac{\partial E_x}{\partial t} = \frac{1}{\varepsilon} \left[\frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} - (J_x + \sigma E_x) \right] \quad (3.9)$$

$$\frac{\partial E_y}{\partial t} = \frac{1}{\varepsilon} \left[\frac{\partial H_x}{\partial z} - \frac{\partial H_z}{\partial x} - (J_y + \sigma E_y) \right] \quad (3.10)$$

$$\frac{\partial E_z}{\partial t} = \frac{1}{\varepsilon} \left[\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - (J_z + \sigma E_z) \right] \quad (3.11)$$

e os campos magnéticos pelas seguintes equações:

$$\frac{\partial H_x}{\partial t} = \frac{1}{\mu} \left[\frac{\partial E_y}{\partial z} - \frac{\partial E_z}{\partial y} - (M_x + \sigma^* H_x) \right] \quad (3.12)$$

$$\frac{\partial H_y}{\partial t} = \frac{1}{\mu} \left[\frac{\partial E_z}{\partial x} - \frac{\partial E_x}{\partial z} - (M_y + \sigma^* H_y) \right] \quad (3.13)$$

$$\frac{\partial H_z}{\partial t} = \frac{1}{\mu} \left[\frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x} - (M_z + \sigma^* H_z) \right] \quad (3.14)$$

A célula de Yee, como a apresentada na Fig. 3.1, é a base para a representação dos campos elétricos e magnéticos no FDTD, pois mostra o acoplamento das equações (3.9) à (3.14) que ocorre através dos rotacionais dos campos elétricos e magnéticos, interligando entre elas nas direções x , y e z . Dessa forma, provê a interligação dos contornos nas Leis de Ampère e Faraday.

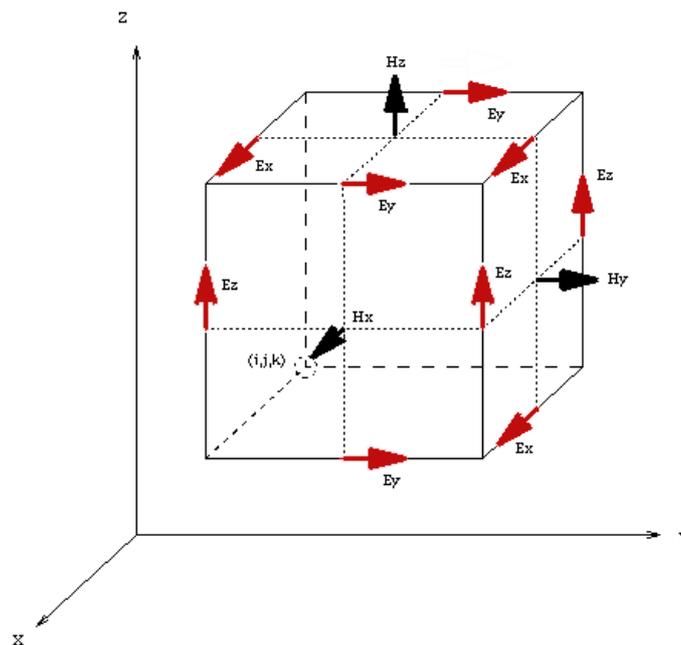


Fig. 3.1 – Esquema de acoplamento dos campos elétricos e magnéticos

Na Fig 3.2 é mostrado que a célula de Yee está centrada nas componentes de tempo de \vec{E} e \vec{H} , que é chamado como arranjo *leapfrog* (salto da rã). Nesse caso, todos os dados processados para a modelagem do espaço \vec{E} são realizados e armazenados na memória em um dado instante de tempo, sendo que eles utilizam dados previamente armazenados em \vec{H} . Então, todos os dados \vec{H} processados e armazenados utilizam os dados \vec{E} já computados. Esse processo continua até que os passos de tempo sejam concluídos, ocorrendo através do *leapfrog time-stepping* que é explícito; desta forma são evitados problemas que envolvam simultaneamente equações e inversões matriciais.

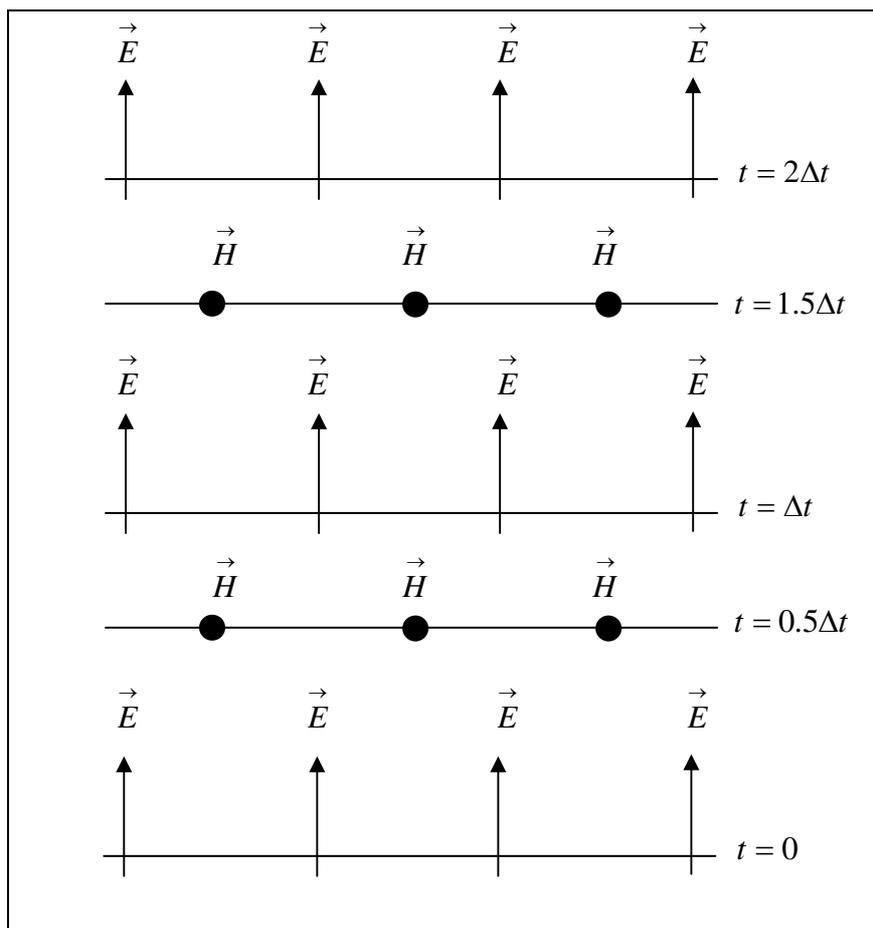


Fig. 3.2 – Esquema *leapfrog time-step*

A precisão dos incrementos espaciais é dada por Δx , Δy e Δz , para as direções x , y e z , sendo que i , j e k são inteiros. Assim, é denotada qualquer função u , de espaço e tempo em um ponto discreto na malha e no tempo como,

$$u(i\Delta x, j\Delta y, k\Delta z, n\Delta t) = u_{i,j,k}^n, \quad (3.15)$$

sendo Δt o incremento no tempo. Substituindo essa notação nas diferenças finitas de espaço e derivadas no tempo, utilizadas por Yee, e considerando a primeira derivada espacial de u na direção x , fixando o tempo em $t_n = n\Delta t$:

$$\frac{\partial u}{\partial x}(i\Delta x, j\Delta y, k\Delta z, n\Delta t) = \frac{u_{i+1/2,j,k}^n - u_{i-1/2,j,k}^n}{\Delta x} + O[(\Delta x)^2] \quad (3.16)$$

Utilizando a notação acima especificada, pode-se discretizar as equações de Maxwell. Porém, de acordo com a formulação do FDTD, em que diferentes intervalos de tempo são necessários numa mesma equação a um mesmo campo, é necessário utilizar o que Taflove e Hagness [38] chamam de “aproximação semi-implícita”, ou média ponderada:

$$E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^n = \frac{E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} + E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n-\frac{1}{2}}}{2} \quad (3.17)$$

A resolução da Eq. (3.9) através da representação espaço-temporal apresentada na Eq. (3.16), pode ser reescrita da seguinte maneira:

$$E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} = \left(\frac{1 - \frac{\sigma_{i,j+\frac{1}{2},k+\frac{1}{2}} \Delta t}{2\varepsilon_{i,j+\frac{1}{2},k+\frac{1}{2}}}}{1 + \frac{\sigma_{i,j+\frac{1}{2},k+\frac{1}{2}} \Delta t}{2\varepsilon_{i,j+\frac{1}{2},k+\frac{1}{2}}}} \right) E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n-\frac{1}{2}} + \left(\frac{\frac{\Delta t}{\varepsilon_{i,j+\frac{1}{2},k+\frac{1}{2}}}}{1 + \frac{\sigma_{i,j+\frac{1}{2},k+\frac{1}{2}} \Delta t}{2\varepsilon_{i,j+\frac{1}{2},k+\frac{1}{2}}}} \right) \cdot \left(\frac{H_z \Big|_{i,j+1,k+\frac{1}{2}}^n - H_z \Big|_{i,j,k+\frac{1}{2}}^n}{\Delta y} - \frac{H_y \Big|_{i,j+\frac{1}{2},k+1}^n - H_y \Big|_{i,j+\frac{1}{2},k}^n}{\Delta z} - J_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^n \right) \quad (3.18)$$

Para os campos elétricos nas outras direções (y e z), o algoritmo discretizado é similar, variando apenas as derivadas de acordo com espaço e tempo.

$$E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} = \left(\frac{1 - \frac{\sigma_{i-\frac{1}{2},j+1,k+\frac{1}{2}} \Delta t}{2\epsilon_{i-\frac{1}{2},j+1,k+\frac{1}{2}}}}{1 + \frac{\sigma_{i-\frac{1}{2},j+1,k+\frac{1}{2}} \Delta t}{2\epsilon_{i-\frac{1}{2},j+1,k+\frac{1}{2}}}} \right) E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n-\frac{1}{2}} + \left(\frac{\frac{\Delta t}{\epsilon_{i-\frac{1}{2},j+1,k+\frac{1}{2}}}}{1 + \frac{\sigma_{i-\frac{1}{2},j+1,k+\frac{1}{2}} \Delta t}{2\epsilon_{i-\frac{1}{2},j+1,k+\frac{1}{2}}}} \right) \cdot \quad (3.19)$$

$$\left(\frac{H_x \Big|_{i-\frac{1}{2},j+1,k+1}^n - H_x \Big|_{i-\frac{1}{2},j+1,k}^n}{\Delta z} - \frac{H_z \Big|_{i,j+1,k+\frac{1}{2}}^n - H_z \Big|_{i-1,j+1,k+\frac{1}{2}}^n}{\Delta x} - J_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^n \right)$$

$$E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} = \left(\frac{1 - \frac{\sigma_{i-\frac{1}{2},j+\frac{1}{2},k+1} \Delta t}{2\epsilon_{i-\frac{1}{2},j+\frac{1}{2},k+1}}}{1 + \frac{\sigma_{i-\frac{1}{2},j+\frac{1}{2},k+1} \Delta t}{2\epsilon_{i-\frac{1}{2},j+\frac{1}{2},k+1}}} \right) E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n-\frac{1}{2}} + \left(\frac{\frac{\Delta t}{\epsilon_{i-\frac{1}{2},j+\frac{1}{2},k+1}}}{1 + \frac{\sigma_{i-\frac{1}{2},j+\frac{1}{2},k+1} \Delta t}{2\epsilon_{i-\frac{1}{2},j+\frac{1}{2},k+1}}} \right) \cdot \quad (3.20)$$

$$\left(\frac{H_y \Big|_{i,j+\frac{1}{2},k+1}^n - H_y \Big|_{i-1,j+\frac{1}{2},k+1}^n}{\Delta x} - \frac{H_x \Big|_{i-\frac{1}{2},j+1,k+1}^n - H_x \Big|_{i-\frac{1}{2},j,k+1}^n}{\Delta y} - J_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^n \right)$$

Similarmente, é possível expressar os campos magnéticos H_x (Eq. 3.12), H_y (Eq. (3.13)) e H_z (Eq. (3.14)) seguindo o mesmo procedimento dos campos elétricos.

$$\begin{aligned}
H_x \Big|_{i-\frac{1}{2},j+1,k+1}^{n+1} &= \left(\frac{1 - \frac{\sigma^*_{i-\frac{1}{2},j+1,k+1} \Delta t}{2\mu_{i-\frac{1}{2},j+1,k+1}}}{1 + \frac{\sigma^*_{i-\frac{1}{2},j+1,k+1} \Delta t}{2\mu_{i-\frac{1}{2},j+1,k+1}}} \right) H_x \Big|_{i-\frac{1}{2},j+1,k+1}^n + \left(\frac{\frac{\Delta t}{\mu_{i-\frac{1}{2},j+1,k+1}}}{1 + \frac{\sigma^*_{i-\frac{1}{2},j+1,k+1} \Delta t}{2\mu_{i-\frac{1}{2},j+1,k+1}}} \right). \\
&\left(\frac{E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{3}{2}}^{n+\frac{1}{2}} - E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta z} - \frac{E_z \Big|_{i-\frac{1}{2},j+\frac{3}{2},k+1}^{n+\frac{1}{2}} - E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}}}{\Delta y} - M_x \Big|_{i-\frac{1}{2},j+1,k+1}^{n+\frac{1}{2}} \right)
\end{aligned} \tag{3.21}$$

$$\begin{aligned}
H_y \Big|_{i,j+\frac{1}{2},k+1}^{n+1} &= \left(\frac{1 - \frac{\sigma^*_{i,j+\frac{1}{2},k+1} \Delta t}{2\mu_{i,j+\frac{1}{2},k+1}}}{1 + \frac{\sigma^*_{i,j+\frac{1}{2},k+1} \Delta t}{2\mu_{i,j+\frac{1}{2},k+1}}} \right) H_y \Big|_{i,j+\frac{1}{2},k+1}^n + \left(\frac{\frac{\Delta t}{\mu_{i,j+\frac{1}{2},k+1}}}{1 + \frac{\sigma^*_{i,j+\frac{1}{2},k+1} \Delta t}{2\mu_{i,j+\frac{1}{2},k+1}}} \right). \\
&\left(\frac{E_z \Big|_{i+\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} - E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}}}{\Delta x} - \frac{E_x \Big|_{i,j+\frac{1}{2},k+\frac{3}{2}}^{n+\frac{1}{2}} - E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta z} - M_y \Big|_{i,j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} \right)
\end{aligned} \tag{3.22}$$

$$\begin{aligned}
H_z \Big|_{i,j,k+\frac{1}{2}}^{n+1} &= \left(\frac{1 - \frac{\sigma^*_{i,j+1,k+\frac{1}{2}} \Delta t}{2\mu_{i,j+1,k+\frac{1}{2}}}}{1 + \frac{\sigma^*_{i,j+1,k+\frac{1}{2}} \Delta t}{2\mu_{i,j+1,k+\frac{1}{2}}}} \right) H_z \Big|_{i,j+1,k+\frac{1}{2}}^n + \left(\frac{\frac{\Delta t}{\mu_{i,j+1,k+\frac{1}{2}}}}{1 + \frac{\sigma^*_{i,j+1,k+\frac{1}{2}} \Delta t}{2\mu_{i,j+1,k+\frac{1}{2}}}} \right). \\
&\left(\frac{E_x \Big|_{i,j+\frac{3}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} - E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta y} - \frac{E_y \Big|_{i+\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} - E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x} - M_z \Big|_{i,j,k+\frac{1}{2}}^{n+\frac{1}{2}} \right)
\end{aligned} \tag{3.23}$$

Os domínios computacionais dos campos eletromagnéticos são formados por muitas células de Yee, como o apresentado na Fig. 3.3. Desta forma, tem-se a exigência de um algoritmo que possua várias matrizes a serem armazenadas, tornando um programa que precisará de razoável quantidade de memória e processamento.

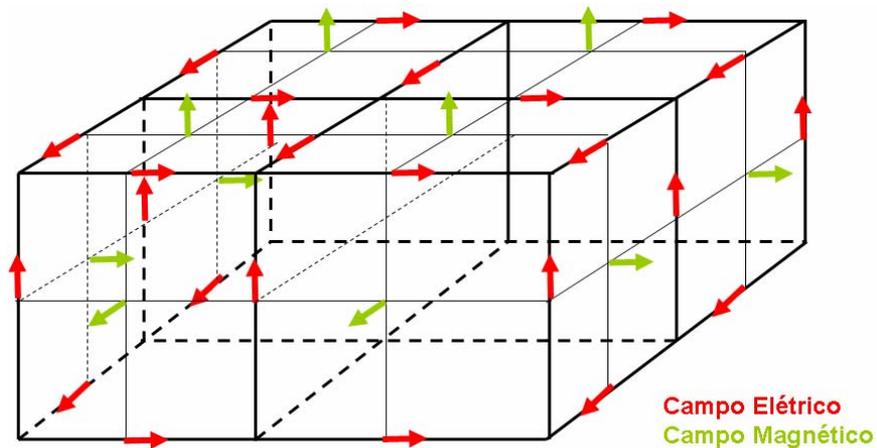


Fig. 3.3 – Quatro Células de Yee interligadas

Porém, existem algumas simplificações interessantes de serem realizadas no método, que melhoram a organização e legibilidade. A organização de códigos numéricos é importante para facilitar a manutenção e a busca pela precisão dos resultados.

Essas melhorias também podem contribuir com o desempenho, desde que sejam devidamente planejadas e validadas. No FDTD, existe uma interessante organização que é possível de ser realizada quando os meios são lineares, explicitando os parâmetros eletromagnéticos ($\sigma, \sigma^*, \epsilon_r$ e μ_r), pois esses valores não variam no tempo. Por esse motivo, os dados são calculados e armazenados em variáveis (espaço de memória) apenas no início do programa, realizando-se isso para cada uma das posições do campo.

Desta forma, as equações de campo elétrico podem ser refeitas como, por exemplo, a discretização de E_x , que pode ser reescrita na forma da Eq. (3.24):

$$E_x \left| \begin{matrix} n+\frac{1}{2} \\ i, j+\frac{1}{2}, k+\frac{1}{2} \end{matrix} \right. = ca_{-ex} \left| \begin{matrix} \\ i, j+\frac{1}{2}, k+\frac{1}{2} \end{matrix} \right. E_x \left| \begin{matrix} n-\frac{1}{2} \\ i, j+\frac{1}{2}, k+\frac{1}{2} \end{matrix} \right. + cb_{-ex} \left| \begin{matrix} \\ i, j+\frac{1}{2}, k+\frac{1}{2} \end{matrix} \right. \cdot \quad (3.24)$$

$$\left(H_z \left| \begin{matrix} n \\ i, j+1, k+\frac{1}{2} \end{matrix} \right. - H_z \left| \begin{matrix} n \\ i, j, k+\frac{1}{2} \end{matrix} \right. + H_y \left| \begin{matrix} n \\ i, j+\frac{1}{2}, k \end{matrix} \right. - H_y \left| \begin{matrix} n \\ i, j+\frac{1}{2}, k+1 \end{matrix} \right. - J_x \left| \begin{matrix} n \\ i, j+\frac{1}{2}, k+\frac{1}{2} \end{matrix} \right. \Delta x \right)$$

sendo:

$$ca_{-ex} \left| \begin{matrix} \\ i, j, k \end{matrix} \right. = \frac{\left(1 - \frac{\sigma_{i,j,k} \Delta t}{2\varepsilon_{i,j,k}} \right)}{\left(1 + \frac{\sigma_{i,j,k} \Delta t}{2\varepsilon_{i,j,k}} \right)} \quad (3.25)$$

$$cb_{-ex} \left| \begin{matrix} \\ i, j, k \end{matrix} \right. = \frac{\left(\frac{\Delta t}{\varepsilon_{i,j,k} \Delta x} \right)}{\left(1 + \frac{\sigma_{i,j,k} \Delta t}{2\varepsilon_{i,j,k}} \right)} \quad (3.26)$$

Dessa forma, as equações de campo elétrico nas direções y (Eq. (3.19)) e z (Eq. (3.20)), também podem ser reescritas seguindo a mesma técnica:

$$E_y \left| \begin{matrix} n+\frac{1}{2} \\ i-\frac{1}{2}, j+1, k+\frac{1}{2} \end{matrix} \right. = ca_{-ey} \left| \begin{matrix} \\ i-\frac{1}{2}, j+1, k+\frac{1}{2} \end{matrix} \right. E_y \left| \begin{matrix} n-\frac{1}{2} \\ i-\frac{1}{2}, j+1, k+\frac{1}{2} \end{matrix} \right. + cb_{-ey} \left| \begin{matrix} \\ i-\frac{1}{2}, j+1, k+\frac{1}{2} \end{matrix} \right. \cdot \quad (3.27)$$

$$\left(H_x \left| \begin{matrix} n \\ i-\frac{1}{2}, j+1, k+1 \end{matrix} \right. - H_x \left| \begin{matrix} n \\ i-\frac{1}{2}, j+1, k \end{matrix} \right. + H_z \left| \begin{matrix} n \\ i-1, j+1, k+\frac{1}{2} \end{matrix} \right. - H_z \left| \begin{matrix} n \\ i, j+1, k+\frac{1}{2} \end{matrix} \right. - J_y \left| \begin{matrix} n \\ i-\frac{1}{2}, j+1, k+\frac{1}{2} \end{matrix} \right. \Delta y \right)$$

$$E_z \left| \begin{matrix} n+\frac{1}{2} \\ i-\frac{1}{2}, j+\frac{1}{2}, k+1 \end{matrix} \right. = ca_{-ez} \left| \begin{matrix} \\ i-\frac{1}{2}, j+\frac{1}{2}, k+1 \end{matrix} \right. E_z \left| \begin{matrix} n-\frac{1}{2} \\ i-\frac{1}{2}, j+\frac{1}{2}, k+1 \end{matrix} \right. + cb_{-ez} \left| \begin{matrix} \\ i-\frac{1}{2}, j+\frac{1}{2}, k+1 \end{matrix} \right. \cdot \quad (3.28)$$

$$\left(H_y \left| \begin{matrix} n \\ i, j+\frac{1}{2}, k+1 \end{matrix} \right. - H_y \left| \begin{matrix} n \\ i-1, j+\frac{1}{2}, k+1 \end{matrix} \right. + H_x \left| \begin{matrix} n \\ i-\frac{1}{2}, j, k+1 \end{matrix} \right. - H_x \left| \begin{matrix} n \\ i-\frac{1}{2}, j+1, k+1 \end{matrix} \right. - J_z \left| \begin{matrix} n \\ i-\frac{1}{2}, j+\frac{1}{2}, k+1 \end{matrix} \right. \Delta z \right)$$

Os campos magnéticos, por sua vez, podem ser simplificados separando, no início do programa, as propriedades dos materiais que estão relacionadas aos campos magnéticos, como no campo magnético na direção x, tendo:

$$da_{-hx}\Big|_{i,j,k} = \frac{\left(1 - \frac{\sigma_{i,j,k}^* \Delta t}{2\mu_{i,j,k}}\right)}{\left(1 + \frac{\sigma_{i,j,k}^* \Delta t}{2\mu_{i,j,k}}\right)} \quad (3.29)$$

$$db_{-hx}\Big|_{i,j,k} = \frac{\left(\frac{\Delta t}{\mu_{i,j,k} \Delta x}\right)}{\left(1 + \frac{\sigma_{i,j,k}^* \Delta t}{2\mu_{i,j,k}}\right)}, \quad (3.30)$$

possibilitando reescrever as equações dos campos magnéticos nas direções x (Eq. (3.21)), y (Eq. (3.22)) e z (Eq. (3.23)) da seguinte maneira:

$$H_x\Big|_{i-\frac{1}{2},j+1,k+1}^{n+1} = da_{-hx}\Big|_{i-\frac{1}{2},j+1,k+1} H_x\Big|_{i-\frac{1}{2},j+1,k+1}^n + db_{-hx}\Big|_{i-\frac{1}{2},j+1,k+1} \cdot$$

$$\left(E_y\Big|_{i-\frac{1}{2},j+1,k+\frac{3}{2}}^{n+\frac{1}{2}} - E_y\Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} + E_z\Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} - E_z\Big|_{i-\frac{1}{2},j+\frac{3}{2},k+1}^{n+\frac{1}{2}} - M_x\Big|_{i-\frac{1}{2},j+1,k+1}^{n+\frac{1}{2}} \Delta x \right) \quad (3.31)$$

$$H_y\Big|_{i,j+\frac{1}{2},k+1}^{n+1} = da_{-hy}\Big|_{i,j+\frac{1}{2},k+1} H_y\Big|_{i,j+\frac{1}{2},k+1}^n + db_{-hy}\Big|_{i,j+\frac{1}{2},k+1} \cdot$$

$$\left(E_z\Big|_{i+\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} - E_z\Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} + E_x\Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} - E_x\Big|_{i,j+\frac{1}{2},k+\frac{3}{2}}^{n+\frac{1}{2}} - M_y\Big|_{i,j+\frac{1}{2},k+1}^{n+\frac{1}{2}} \Delta y \right) \quad (3.32)$$

$$H_z\Big|_{i,j+1,k+\frac{1}{2}}^{n+1} = da_{-hz}\Big|_{i,j+1,k+\frac{1}{2}} H_z\Big|_{i,j+1,k+\frac{1}{2}}^n + db_{-hz}\Big|_{i,j+1,k+\frac{1}{2}} \cdot$$

$$\left(E_x\Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} - E_x\Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} + E_y\Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} - E_y\Big|_{i+\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} - M_z\Big|_{i,j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} \Delta z \right) \quad (3.33)$$

A discretização das equações de Maxwell ainda pode ser simplificada de acordo com a não propagação das ondas em uma das três dimensões (direções), tornando-se uma solução de duas dimensões. Para exemplificar essa afirmação, Peterson, Ray e Mittra [39] apresentam isso através de uma

estrutura cilíndrica na qual um dos campos não varia ao longo de um dos eixos da geometria; eles, também, levam em consideração as propriedades do material com um certo ε_r, μ_r . Caso o eixo do cilindro se alongue pelo eixo z, no sistema de coordenadas cartesianas, é conveniente separar os campos em transverso magnético (TM) e transverso elétrico (TE), com a variação podendo ser em z. Nesse caso, a componente \hat{z} do campo magnético não está presente no caso TM e, no modo TE, a componente \hat{z} não existe no campo elétrico. O equacionamento do campo eletromagnético, no modo TM_z , pode ser escrito da seguinte forma [38]:

$$\frac{\partial H_x}{\partial t} = \frac{1}{\mu} \left[-\frac{\partial E_z}{\partial y} - (M_x + \sigma * H_x) \right] \quad (3.34)$$

$$\frac{\partial H_y}{\partial t} = \frac{1}{\mu} \left[\frac{\partial E_z}{\partial x} - (M_y + \sigma * H_y) \right] \quad (3.35)$$

$$\frac{\partial E_z}{\partial t} = \frac{1}{\varepsilon} \left[\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - (J_z + \sigma E_z) \right] \quad (3.36)$$

enquanto que o caso TE_z pode ser:

$$\frac{\partial E_x}{\partial t} = \frac{1}{\varepsilon} \left[\frac{\partial H_z}{\partial y} - (J_{fontex} + \sigma E_x) \right] \quad (3.37)$$

$$\frac{\partial E_y}{\partial t} = \frac{1}{\varepsilon} \left[-\frac{\partial H_z}{\partial x} - (J_{fontey} + \sigma E_y) \right] \quad (3.38)$$

$$\frac{\partial H_z}{\partial t} = \frac{1}{\mu} \left[\frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x} - (M_z + \sigma * H_z) \right] \quad (3.39)$$

Os algoritmos referentes aos modos TM_z e TE_z são apresentados da seguinte maneira:

Modo TM_z :

$$E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} = ca_{-} ez \Big|_{i-\frac{1}{2},j+\frac{1}{2}}^{n-\frac{1}{2}} E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2}}^{n-\frac{1}{2}} + cb_{-} ez \Big|_{i-\frac{1}{2},j+\frac{1}{2}}^{n-\frac{1}{2}} \cdot$$

$$\left(H_y \Big|_{i,j+\frac{1}{2}}^n - H_y \Big|_{i-1,j+\frac{1}{2}}^n + H_x \Big|_{i-\frac{1}{2},j+1}^n - H_x \Big|_{i-\frac{1}{2},j+1}^n - J_{fontez} \Big|_{i-\frac{1}{2},j+\frac{1}{2}}^n \Delta z \right)$$
(3.40)

$$H_x \Big|_{i-\frac{1}{2},j+1}^{n+1} = da_{-} hx \Big|_{i-\frac{1}{2},j+1}^n H_x \Big|_{i-\frac{1}{2},j+1}^n + db_{-} hx \Big|_{i-\frac{1}{2},j+1}^n \cdot$$

$$\left(E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} - E_z \Big|_{i-\frac{1}{2},j+\frac{3}{2}}^{n+\frac{1}{2}} - M_{fontex} \Big|_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} \Delta x \right)$$
(3.41)

$$H_y \Big|_{i,j+\frac{1}{2}}^{n+1} = da_{-} hy \Big|_{i,j+\frac{1}{2}}^n H_y \Big|_{i,j+\frac{1}{2}}^{n+1} + db_{-} hz \Big|_{i,j+\frac{1}{2}}^{n+1} \cdot$$

$$\left(E_z \Big|_{i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} - E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} - M_{fontey} \Big|_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} \Delta y \right)$$
(3.42)

Modo TE_z :

$$E_x \Big|_{i,j+\frac{1}{2}}^{n+\frac{1}{2}} = ca_{-} ex \Big|_{i,j+\frac{1}{2}}^{n-\frac{1}{2}} E_x \Big|_{i,j+\frac{1}{2}}^{n-\frac{1}{2}} + cb_{-} ex \Big|_{i,j+\frac{1}{2}}^{n-\frac{1}{2}} \cdot$$

$$\left(H_z \Big|_{i,j+1}^n - H_z \Big|_{i,j}^n - J_{fontex} \Big|_{i,j+\frac{1}{2}}^n \Delta x \right)$$
(3.43)

$$E_y \Big|_{i-\frac{1}{2},j+1}^{n+\frac{1}{2}} = ca_{-} ey \Big|_{i-\frac{1}{2},j+1}^{n-\frac{1}{2}} E_y \Big|_{i-\frac{1}{2},j+1}^{n-\frac{1}{2}} + cb_{-} ey \Big|_{i-\frac{1}{2},j+1}^{n-\frac{1}{2}} \cdot$$

$$\left(H_z \Big|_{i-1,j+1}^n - H_z \Big|_{i,j+1}^n - J_{fontey} \Big|_{i-\frac{1}{2},j+1}^n \Delta y \right)$$
(3.44)

$$H_z \Big|_{i,j+1}^{n+1} = da_{-} hz \Big|_{i,j+1}^n H_z \Big|_{i,j+1}^n + db_{-} hz \Big|_{i,j+1}^{n+1} \cdot$$

$$\left(E_x \Big|_{i,j+\frac{3}{2}}^{n+\frac{1}{2}} - E_x \Big|_{i-\frac{1}{2},j+1}^{n+\frac{1}{2}} + E_y \Big|_{i-\frac{1}{2},j+1}^{n+\frac{1}{2}} - E_y \Big|_{i+\frac{1}{2},j+1}^{n+\frac{1}{2}} - M_{fontez} \Big|_{i,j+1}^{n+\frac{1}{2}} \Delta z \right)$$
(3.45)

Para finalizar essa descrição, segue as afirmações de Araújo *et al.* [40], que considera o FDTD uma das técnicas numéricas mais utilizadas no eletromagnetismo computacional, pois apresenta credibilidade e precisão nos

resultados computacionais, além de satisfazer a solução de muitos problemas. No entanto, o método requer grande quantidade de memória e o tempo de processamento é longo.

3.4 Camadas Perfeitamente Casadas – Perfectly Matched Layer (PML)

Para Moore *et al.* [41], o uso de técnicas finitas com operadores para radiações, nos domínios do tempo e da frequência, é adequado à criação de condições de contorno da radiação (RBC), permitindo truncar o domínio computacional volumétrico eletricamente fechado, para modelar de acordo com os objetivos e simular de maneira eficiente a extensão do domínio computacional até o infinito. As condições de contorno de radiação, devido aos seus objetivos de absorver as ondas eletromagnéticas que chegam as suas bordas, também recebem o nome de condições de contorno de absorção (ABCs). Com esse recurso, o domínio computacional Ω pode ser representado de acordo com a Fig. 3.4, com sua respectiva condição de contorno $\partial\Omega$, para simular a extensão da malha no infinito.

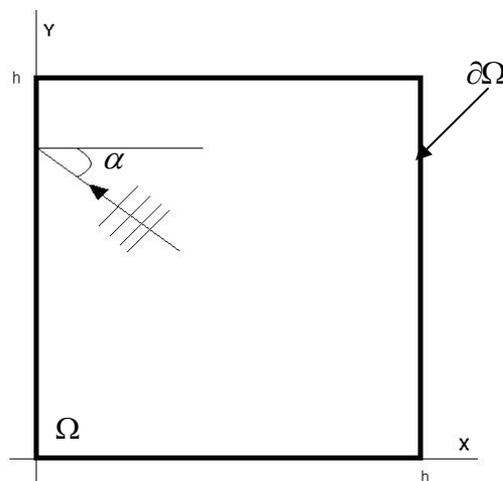


Fig. 3.4 – Domínio computacional Ω cartesiano e bidimensional, apresentado por Moore [41].

Bérenger [42], no ano de 1994, publicou em seu trabalho uma técnica ABC para a simulação de espaço livre, a qual ele se referiu como PML (*Perfect Matched Layer*). Essa técnica é baseada no uso de camadas especificamente projetadas para absorver, sem reflexão, as ondas eletromagnéticas. As preocupações de Bérenger recaiam sobre casos particulares como, por

exemplo, se o plano e a propagação fossem perpendiculares às bordas, isso implicava em imperfeições que impossibilitavam o tratamento de alguns problemas e impunham restrições em outros. A PML de Bérenger, como é conhecida, possui em seu meio um fator teórico de reflexão de uma onda plana sobre uma interface com várias camadas como sendo nulo, em qualquer frequência e ângulo de incidência, contrariamente ao meio em que tal fator é nulo somente em uma incidência normal. Esse recurso numérico tem a mesma funcionalidade dos materiais absorvedores das câmaras anecóicas, como a apresentada na Fig. 3.5.

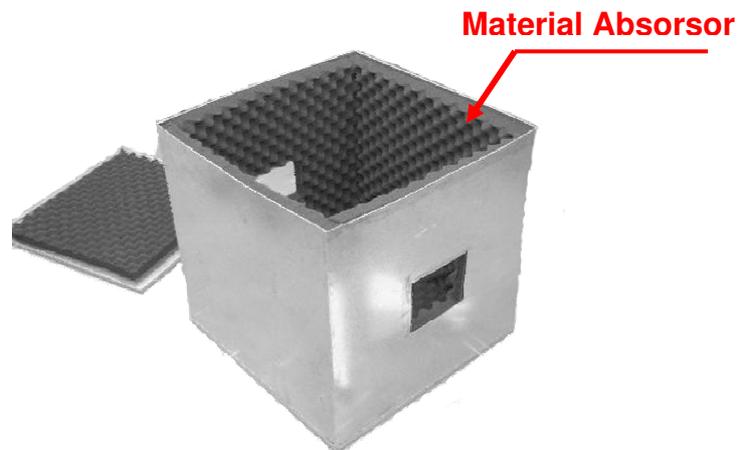


Fig. 3.5 – Mini-câmara anecóica construída por Sartori para seus experimentos com metamateriais [43]

O emprego da PML só é realizável após à satisfação da seguinte condição:

$$\frac{\sigma}{\varepsilon_0} = \frac{\sigma^*}{\mu_0} \quad (3.46)$$

sendo a impedância do meio igual a do vácuo e caso não ocorra reflexão quando uma onda plana normal atravessa normalmente uma interface entre um meio e o vácuo. No caso tridimensional, o equacionamento de todas as seis componentes vetoriais do campo cartesiano é dividido por Katz *et al.* [44], resultando nas adaptações das equações de Maxwell para a implementação da PML, sendo as doze equações:

Campos elétricos divididos

$$\left(\varepsilon \frac{\partial}{\partial t} + \sigma_y \right) E_{xy} = \frac{\partial}{\partial y} (H_{zx} + H_{zy}) \quad (3.47)$$

$$\left(\varepsilon \frac{\partial}{\partial t} + \sigma_z \right) E_{xz} = -\frac{\partial}{\partial z} (H_{yx} + H_{yz}) \quad (3.48)$$

$$\left(\varepsilon \frac{\partial}{\partial t} + \sigma_z \right) E_{yz} = \frac{\partial}{\partial z} (H_{xy} + H_{xz}) \quad (3.49)$$

$$\left(\varepsilon \frac{\partial}{\partial t} + \sigma_x \right) E_{yx} = -\frac{\partial}{\partial x} (H_{zx} + H_{zy}) \quad (3.50)$$

$$\left(\varepsilon \frac{\partial}{\partial t} + \sigma_x \right) E_{zx} = \frac{\partial}{\partial x} (H_{yx} + H_{yz}) \quad (3.51)$$

$$\left(\varepsilon \frac{\partial}{\partial t} + \sigma_y \right) E_{zy} = -\frac{\partial}{\partial y} (H_{xy} + H_{xz}) \quad (3.52)$$

Campos magnéticos divididos

$$\left(\mu \frac{\partial}{\partial t} + \sigma_y^* \right) H_{xy} = -\frac{\partial}{\partial y} (E_{zx} + E_{zy}) \quad (3.53)$$

$$\left(\mu \frac{\partial}{\partial t} + \sigma_z^* \right) H_{xz} = \frac{\partial}{\partial z} (E_{yx} + E_{yz}) \quad (3.54)$$

$$\left(\mu \frac{\partial}{\partial t} + \sigma_z^* \right) H_{yz} = -\frac{\partial}{\partial z} (E_{xy} + E_{xz}) \quad (3.55)$$

$$\left(\mu \frac{\partial}{\partial t} + \sigma_x^* \right) H_{yx} = \frac{\partial}{\partial x} (E_{zx} + E_{zy}) \quad (3.56)$$

$$\left(\mu \frac{\partial}{\partial t} + \sigma_x^* \right) H_{zx} = -\frac{\partial}{\partial x} (E_{yx} + E_{yz}) \quad (3.57)$$

$$\left(\mu \frac{\partial}{\partial t} + \sigma_y^* \right) H_{zy} = \frac{\partial}{\partial y} (E_{xy} + E_{xz}) \quad (3.58)$$

A Fig. 3.6 mostra um exemplo de um guia ondas retangular com um objeto dielétrico posicionado no domínio computacional, tendo nas extremidades, em z , as PMLs condicionando o final problema.

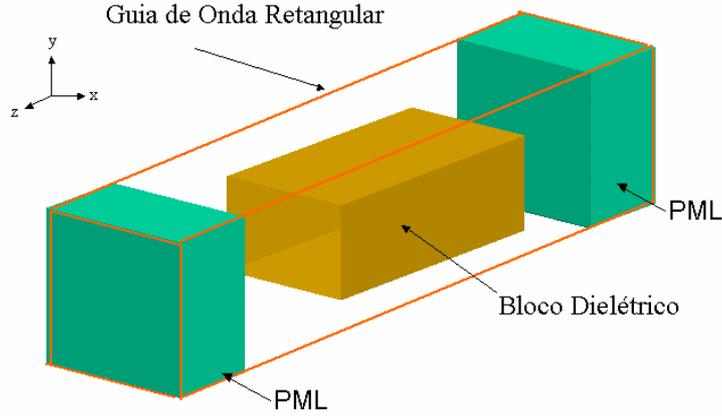


Fig. 3.6 – Exemplo de uso da PML num guia de onda retangular

Do modo em que é definido o transversal elétrico com relação a z (TE_z) o equacionamento necessário à implementação da PML realiza o *split-field* no campo H_z , como é apresentado por Berenger [45]:

$$\varepsilon_0 \frac{\partial E_x}{\partial t} + \sigma_y E_x = \frac{\partial(H_{zx} + H_{zy})}{\partial y} \quad (3.59)$$

$$\varepsilon_0 \frac{\partial E_y}{\partial t} + \sigma_x E_y = \frac{\partial(H_{zx} + H_{zy})}{\partial x} \quad (3.60)$$

$$\mu_0 \frac{\partial H_{zx}}{\partial t} + \sigma_x^* H_{zx} = -\frac{\partial E_y}{\partial x} \quad (3.61)$$

$$\mu_0 \frac{\partial H_{zy}}{\partial t} + \sigma_y^* H_{zy} = \frac{\partial E_x}{\partial y} \quad (3.62)$$

Neste caso as condutividades elétricas e magnéticas devem satisfazer a condição de acoplamento da impedância, apresentada na Eq. (3.46). No modo TM_z , o equacionamento, através da técnica *split-field*, possibilita reescrever as equações da seguinte maneira:

$$\mu_0 \frac{\partial H_x}{\partial t} + \sigma_x^* H_x = -\frac{\partial(E_{zx} + E_{zy})}{\partial y} \quad (3.63)$$

$$\mu_0 \frac{\partial H_y}{\partial t} + \sigma_y^* H_y = -\frac{\partial(E_{zx} + E_{zy})}{\partial x} \quad (3.64)$$

$$\varepsilon_0 \frac{\partial E_{zx}}{\partial t} + \sigma_x E_{zx} = \frac{\partial H_y}{\partial x} \quad (3.65)$$

$$\varepsilon_0 \frac{\partial E_{zy}}{\partial t} + \sigma_y E_{zy} = -\frac{\partial H_x}{\partial y} \quad (3.66)$$

3.5 FDTD em Paralelo

A computação paralela, como já definida, consiste da execução de vários processos, num dado momento para a resolução de um único problema. Quando é utilizado um *cluster Beowulf* com várias estações de trabalho, as tarefas são divididas entre essas máquinas, procurando agilizar e, na maioria dos casos, possibilitar que problemas de médio e grande porte sejam resolvidos sem a necessidade aquisitiva de um supercomputador.

Nesse trabalho, optou-se pela paralelização do método FDTD dividindo o domínio computacional em apenas uma das direções, que neste caso foi a x . Na Fig. 3.7 é apresentado um esquema dessa divisão do domínio tridimensional do método em três processos.

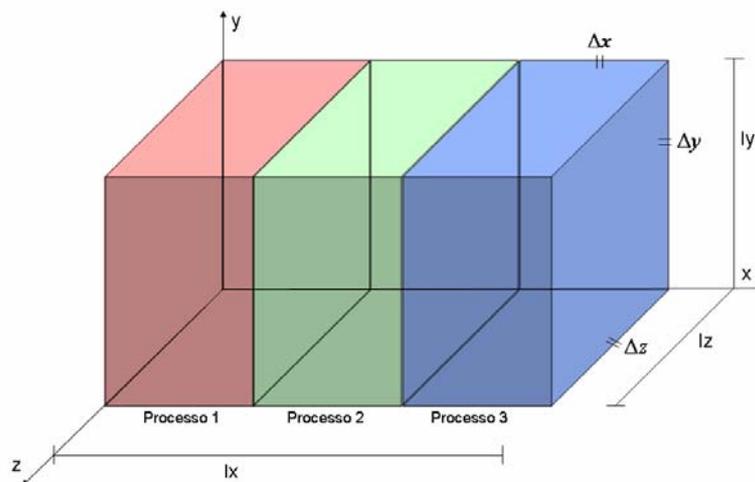


Fig. 3.7 – Esquema de paralelização do domínio computacional tridimensional do FDTD em três processos.

Com a divisão do domínio computacional surgem as fronteiras dos processos, causadas devido à necessidade das trocas de dados através de pacotes, possibilitando que o programa paralelo conclua a simulação corretamente. Na Fig. 3.7 é mostrado o esquema de distribuições dos campos eletromagnéticos nos processos, através da segmentação do domínio na direção x , implicando na troca dos dados referentes aos campos E_y, E_z, H_y e

H_z . As trocas de dados são realizadas através de mensagens contendo matrizes de dimensões $(l_y \times l_z)$, ou seja, uma única linha inteira da matriz tridimensional $(l_x \times l_y \times l_z)$ fica sendo referente à fronteira do processo. As trocas de mensagens estão representadas na Fig. 3.8, que indica os pacotes no padrão MPI enviados de um processo ao outro.

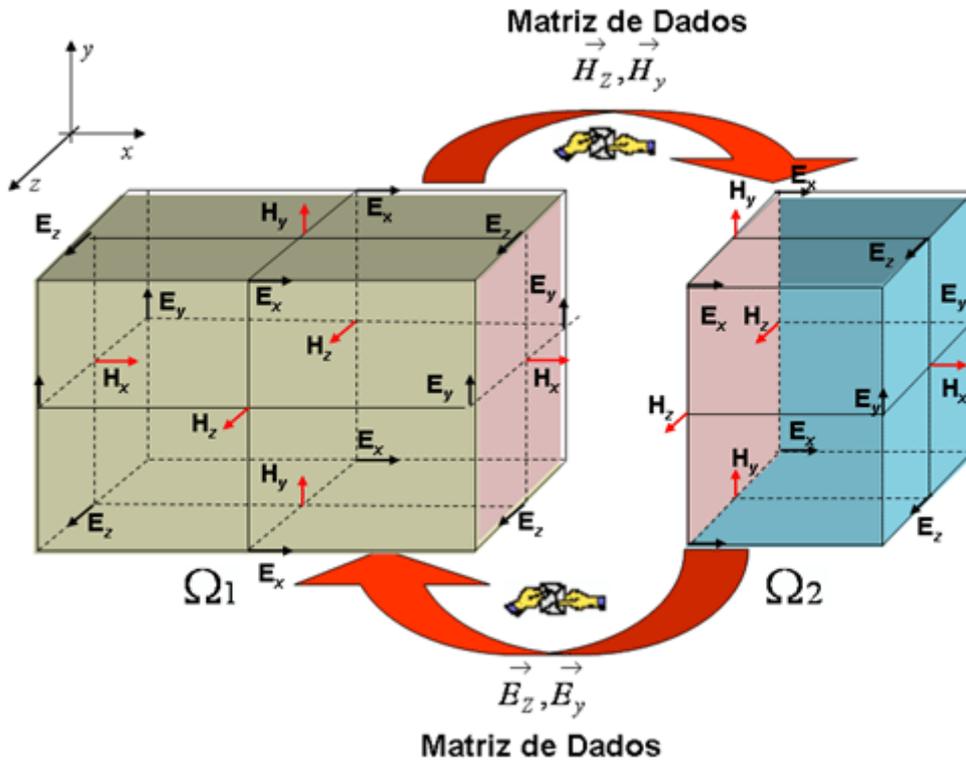


Fig. 3.8 – Troca de pacotes entre dois processos

As trocas de mensagens entre os processos podem ser realizadas através das bibliotecas *Parallel Virtual Machine* (PVM) [46] ou através do padrão *Message Passing Interface* (MPI) [47], utilizando alguma biblioteca específica como, por exemplo, LAM-MPI, MPICH, ou qualquer outra. Neste trabalho optou-se pela LAM-MPI, devido à sua gratuidade, desempenho e documentação disponível na Internet. No padrão MPI, o funcionamento das trocas de mensagens é simples, pois uma primitiva de envio (*SEND*) persistente envia os dados necessários de um processo ao outro. Enquanto isso, o outro processo aciona uma primitiva de recebimento (*RECV*) da mensagem. Na Fig. 3.8 é possível notar que os pacotes do “Processo 1” são enviados ao “Processo 2” e vice-versa, sendo necessários comandos que validem o recebimento. Para exemplificar, a mensagem contendo a matriz H_z

com o conteúdo da fronteira do “Processo 1” é enviada ao “Processo 2”, que em determinado momento aciona a primitiva de recebimento da mensagem.

O desenvolvimento desse tipo de algoritmo paralelo é considerado portátil, devido à independência da arquitetura física dos computadores, adaptabilidade em alguns sistemas operacionais como, Linux e Microsoft Windows e, por fim, ao desenvolvimento de programas em diferentes linguagens de programação que se comunicam (trocam mensagens) normalmente. Nesse trabalho foi utilizado o sistema operacional Linux, nas distribuições RedHat 9.0 e Fedora (Core 2 e 3), com a biblioteca LAM-MPI para as trocas de mensagens e o compilador *gcc* para geração dos programas.

Essa estrutura de desenvolvimento foi adotada, previamente, no desenvolvimento dos códigos FDTD em duas dimensões e paralelos, porém os dados trocados são vetores linhas. Na Fig. 3.9 é apresentado o esquema de paralelização implementado por Araújo [40], o qual foi tomado como base para o desenvolvimento do algoritmo paralelo utilizado em simulações de propagação de ondas eletromagnéticas, no modo TE_z .

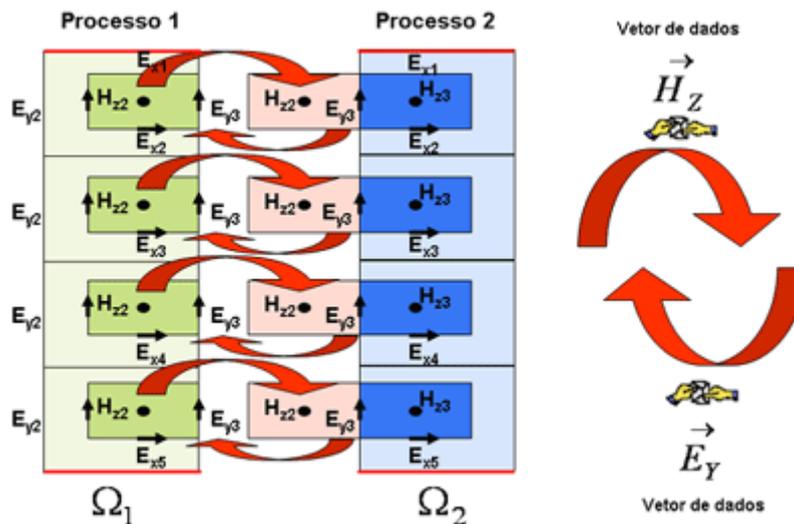


Fig. 3.9 – Troca de mensagens entre processos do FDTD paralelo no modo TE_z .

Nesse caso, um vetor referente ao campo eletromagnético H_z , na fronteira da “Máquina 1”, é enviado à “Máquina 2” por uma mensagem. Outra mensagem é passada da “Máquina 2” à “Máquina 1”, tendo como conteúdo um vetor referente à fronteira do campo E_y .

Em resumo, quando o domínio computacional do método das diferenças finitas, no domínio do tempo, é dividido para ser distribuído entre os processos

em execução no *cluster*, surgem as fronteiras dos processos que devem ser trocadas através de mensagens que possibilitam a obtenção da solução. No Anexo B é apresentado, de uma maneira resumida, o uso da biblioteca LAM-MPI empregada na troca de mensagens entre os processos. No Capítulo 4 são apresentados os algoritmos desenvolvidos, com seus respectivos resultados de desempenho, além da apresentação de algumas aplicações do método FDTD que foram implementadas e testadas. Também, no próximo capítulo, são mencionadas as dificuldades e uma breve avaliação crítica do trabalho.

Capítulo 4

Desenvolvimento e Discussão dos Resultados

4.1 Introdução

O desenvolvimento deste trabalho optou pelo uso de *software* gratuito e aberto, devido à redução nos custos e para seguir os fundamentos do *cluster Beowulf*, que intenta incrementar a capacidade de processamento com um baixo valor financeiro.

Além disso, alguns outros fatores também contribuíram com a escolha e a utilização das ferramentas, dentre as quais se destacam a adaptabilidade dos aplicativos às necessidades, o desempenho, a diversidade de uso por outras instituições, documentação de fácil acesso e pelas atuais propostas incentivadoras do Governo Federal.

A adaptabilidade dos aplicativos é causada pela sua estrutura aberta, ou seja, em que se tem acesso ao código fonte gerador do programa. Isso implica em facilidades aos usuários para que lhes sejam permitidos alterar os códigos fonte dos programas de acordo com suas necessidades, podendo contribuir com o bom desempenho da solução. O desempenho em questão é o tempo de espera para a obtenção de resultados nas simulações computacionais.

Por esses motivos, o *software* livre é um assunto que a cada dia aumenta de proporção, pois surgem novas sociedades, comunidades e grupos que se interessam pelo assunto e pretendem desenvolver algum trabalho relacionado. Para organizar esse desenvolvimento, existe a Licença Pública Geral (GPL) [48], que garante a legitimidade dos aplicativos gratuitos. Caso ocorra algum problema de violação, existem procedimentos a serem tomados que estão especificados no sítio oficial das normas [48]. O desenvolvimento de *software*,

seguindo a GPL, vem sendo adotado por várias instituições públicas e privadas devido às suas possibilidades de uso e baixo custo.

Impulsionado pelas positivas afirmações descritas e os incentivos governamentais na área, a opção pelo *software* livre é importante e estrategicamente eficiente. Outras indiscutíveis questões que favorecem o uso dessas tecnologias, nesse trabalho, foram os bons resultados no ganho de desempenho computacional para simulações de médio e grande porte, utilizando o método FDTD, sem a necessidade de obtenção de novos equipamentos e pacotes de aplicativos que encareceriam o projeto. Assim, tem-se que o ganho de desempenho foi obtido sem custos adicionais em *hardware* e *software*, provendo uma solução que atendeu a demanda computacional e satisfaz os requisitos de estabilidade numérica e precisão dos resultados das simulações numéricas das propriedades eletromagnéticas.

Nesse capítulo são apresentados os resultados dos testes de desempenho dos diferentes *clusters Beowulf* implementados, tendo sido obtidos de simulações de duas e três dimensões de cavidades ressonantes. Também foram exploradas simulações 2D de SAR na cabeça humana e estruturas com metamateriais; desta forma, aplicou-se o algoritmo FDTD paralelo em estudos atuais de grande repercussão mundial.

4.2 Estruturas de Desenvolvimento e Testes

Para a realização desse trabalho duas diferentes locações foram disponibilizadas para as implementações dos *clusters Beowulf*. Os dois laboratórios são vinculados ao Departamento de Microonda e Óptica (DMO) da Faculdade de Engenharia Elétrica e de Computação (FEEC) da Unicamp, sendo o LE-45 e o Laboratório de Microonda Prof. Dr. Rui Fragassi Souza, também chamado de AEL (*Advanced Electromagnetic Laboratory*), o qual foi a base do trabalho e possibilitou diferentes implementações de *clusters*.

4.2.1 Iniciando as Implementações e Analisando os Resultados

No LE-45 foi desenvolvido o primeiro *cluster* para a implementação de programas e simulações paralelos, a fim de explorar os possíveis ganhos de

desempenho utilizando essa técnica. Na Fig. 4.1 é mostrada a interconexão dos equipamentos, tendo sido esta a primeira configuração realizada para as análises.

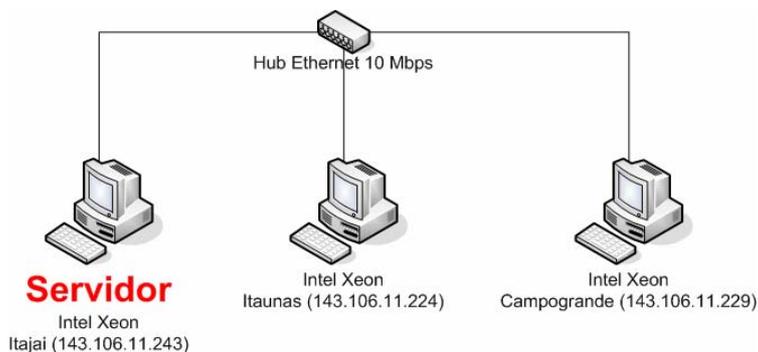


Fig. 4.1 – Primeira estrutura de Cluster da LE-45

Os testes nessa arquitetura visaram exclusivamente confirmar os acréscimos de desempenho que a computação massivamente paralela, utilizando um *cluster* de computadores, poderia trazer. Para isso, foram realizados testes de validação do paralelismo dos códigos, seguindo procedimentos propostos por Laine [49], o qual afirma que devem ser descritas as quantidades de dados a serem processados, o número de nós envolvidos na solução do problema, as características da rede de interconexão, o tipo de *switch* utilizado, além de vários outros dados necessários. Diversas atividades são necessárias à obtenção de resultados concisos sobre a eficiência do sistema, fazendo com que o trabalho seja dispendioso de ser realizado. Na etapa inicial de análises foi utilizado o sistema operacional Linux, da distribuição RedHat 9.0, tendo como base de desenvolvimento o pacote GCC (*GNU Compiler Collection*) [50].

Para a troca de mensagens entre os processos, optou-se pelo padrão MPI (*Message Passing Interface*), o qual surgiu durante o *Workshop on Standards for Messaging Passing in a Distributed Memory Environment*, ocorrido no *Center for Research on Parallel Computing* no ano de 1992 em Williamsburg, Virgínia [52]. A partir desse evento todos os anos são realizados congressos para a discussão sobre esse padrão, que hoje se encontra na versão 2.0. O padrão MPI é utilizado por várias bibliotecas de troca de mensagens, que são mantidas por diferentes grupos como, por exemplo, a LAM-MPI que é atualmente de responsabilidade de um grupo de professores e alunos da

Universidade de Indiana, nos Estados Unidos [53]. A opção pela biblioteca é causada pela disponibilidade gratuita na Internet, por ser eficiente e possuir boa documentação de fácil acesso. No Anexo B, é apresentado um resumo sobre a utilização da LAM-MPI, com um simples exemplo para o processamento paralelo.

Outro fator para a escolha da biblioteca foi a portabilidade, que possibilita desenvolver aplicativos nas diferentes linguagens C, C++ e Fortran, além de permitir que *softwares* desenvolvidos nessas linguagens se comuniquem. Com isso, tem-se um pacote de ferramentas gratuitas utilizadas para o processamento massivamente paralelo, tendo como base o sistema operacional Linux (distribuição RedHat 9.0) e o padrão MPI, através da biblioteca LAM-MPI, utilizando a linguagem C.

A estrutura da rede, como já apresentada na Fig. 4.1, é composta por um Hub Ethernet 10 mbps (modelo SuperStack II) e por três computadores Intel Xeon II de 1 GHz, como 2 GB de memória RAM DDR.

Para a execução dos testes de desempenho no *cluster*, Laine [49] recomenda que sejam feitos programas que calculem matrizes como, por exemplo, a multiplicação de vetor por matriz. Os testes basearam-se nas amostragens de execuções dos programas que variam as dimensões do vetor e da matriz, como segue abaixo:

$$\text{Vetor}_{1 \times n} * \text{Matriz}_{n \times n},$$

tendo n assumido os seguintes valores:

Tabela 4.1 – Valores n para dimensionamento das simulações

5×10^2	1×10^3	2×10^3	3×10^3	4×10^3	5×10^3	6×10^3
7×10^3	8×10^3	9×10^3	10×10^3	11×10^3	12×10^3	13×10^3
14×10^3	15×10^3	16×10^3	17×10^3	18×10^3	19×10^3	20×10^3
21×10^3	22×10^3	23×10^3	24×10^3	25×10^3		

As investigações sobre os tempos gastos para os testes realizados, sobre as diversas condições, estão detalhadas na Fig. 4.2., onde são mostrados os nomes das máquinas utilizadas e as n dimensões.

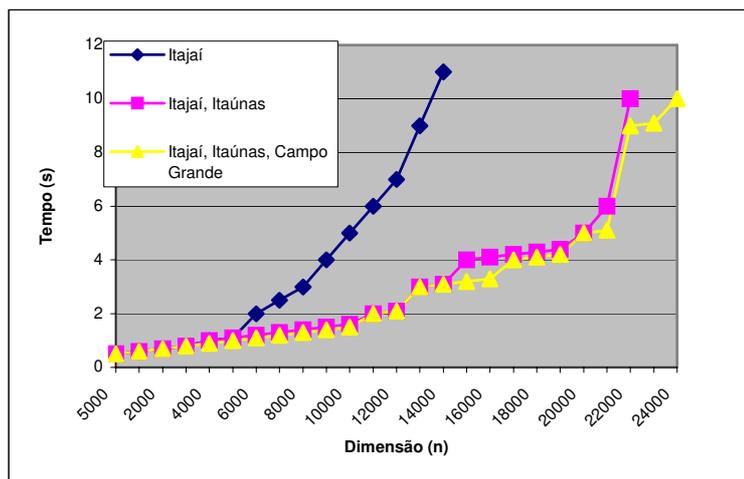


Fig. 4.2 – Testes de desempenho através da multiplicação vetor x matriz

Nesse gráfico, os resultados mostram claramente o ganho de desempenho do problema dividido em duas partes, ou seja, dois processos. Porém, quando foi acrescentada a terceira máquina, o ganho no tempo não foi muito acentuado, ocorrendo vantagens apenas quanto houve possibilidade de aumento do domínio computacional, ou seja, a multiplicação de vetores e matrizes maiores.

Esse resultado apresenta uma razoável eficiência, considerando a experiência adquirida até o momento. Dessa forma, fortalece-se a necessidade de aprofundamento nas investigações e trabalhos necessários ao desenvolvimento de códigos FDTD paralelos eficientes e que atendam as exigências dos grupos de pesquisa do departamento. Na etapa seguinte, os esforços concentraram-se justamente na implementação desse método.

4.2.2 Desenvolvimento e Resultados de Desempenho do FDTD 2D e 3D em Paralelo

A evolução da qualidade de desempenho dos algoritmos paralelos é um processo contínuo, pois existem diversas possibilidades de construções proporcionadas pelos diferentes tipos de uso e estruturas computacionais disponíveis.

Os primeiros códigos desenvolvidos, para solucionar o FDTD, primavam-se pela eficiência e adaptabilidade, pois não existia um número determinado de equipamentos que poderiam ser disponibilizados para a construção do *cluster*.

Com isso, as atividades eram realizadas em uma máquina e emulava-se o uso de vários processos nesta.

Após a construção do *cluster* e as primeiras execuções dos códigos paralelos, a distribuição do processamento ocorria apenas nos cálculos, sendo que todo o domínio era definido nas máquinas. Esse fato ocorreu devido à indeterminação da estrutura em que seriam executados os programas e, principalmente pela falta de experiência. Mesmo assim, os resultados foram significativos, reduzindo bastante o tempo despendido nas simulações. Essa afirmação pode ser notada na Fig. 4.3, que apresenta o gráfico dos testes de desempenho do código FDTD 2D que simula uma cavidade ressonante, que tem as condições de contorno especificadas pela PML de Bérenger.

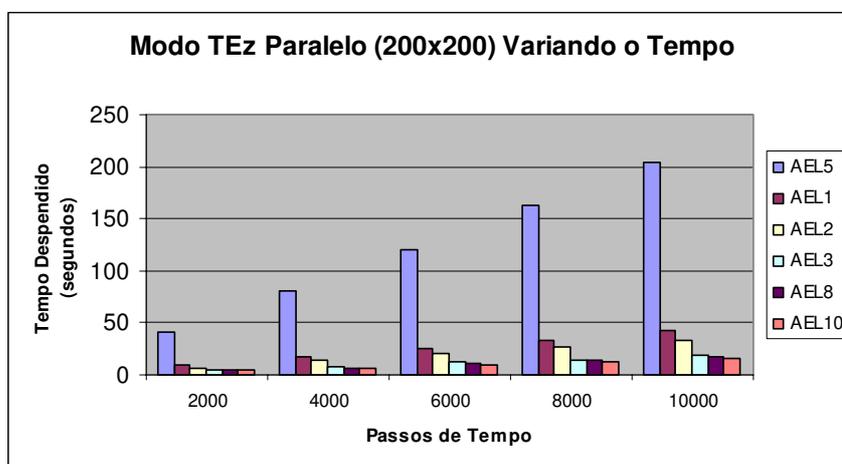


Fig. 4.3 – Gráfico com o ganho computacional obtido no primeiro cluster do AEL.

Nota-se, no gráfico da Fig. 4.3, que o ganho de desempenho extrapola os limites estabelecidos pela Lei de Amdahl (ver Anexo C), que é figura de mérito nesse trabalho. Esse resultado é decorrência da estrutura do *cluster Beowulf* construído (Fig. 4.4), o qual possui máquinas com arquiteturas diferentes e quantidade de recursos variados, onde estabeleceu-se o computador com arquitetura e recursos mais limitados como sendo o servidor do sistema.

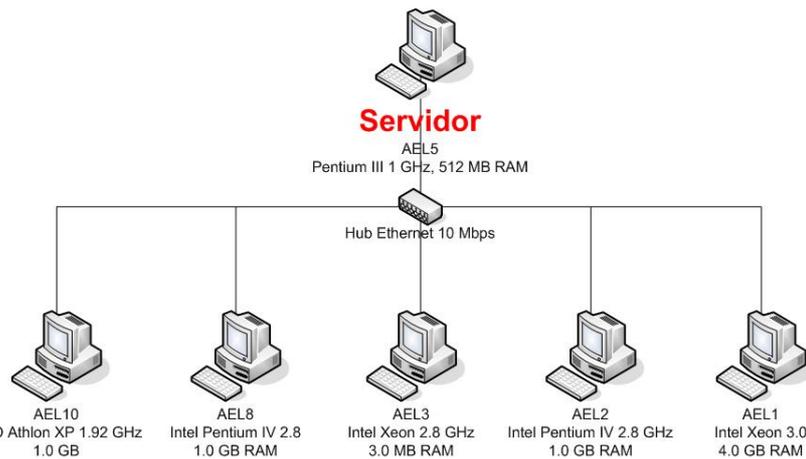


Fig. 4.4 – Esquema do primeiro Cluster montado no AEL (CLUSTER-AEL-1)

A simulação realizada tem a finalidade de apresentar o funcionamento de uma caixa ressonante, com variações temporais. Na Fig. 4.5 são apresentados resultados obtidos em diferentes intervalos de tempo nas simulações da cavidade ressonante, que tem como condições de contorno a PML de Bérenger e uma fonte senoidal excitando o campo E_y .

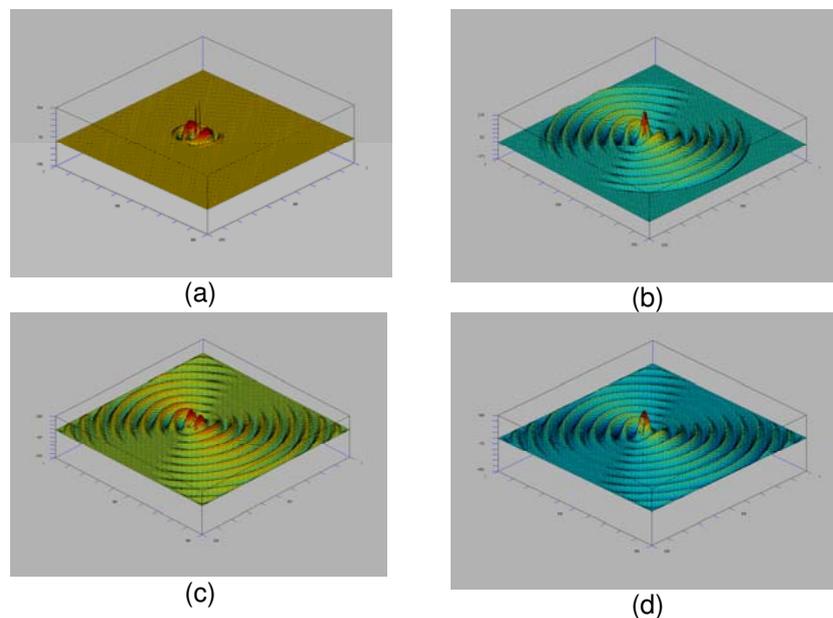


Fig. 4.5 – Resultados da cavidade ressonante em diferentes intervalos de tempo (a) 100 passos, (b) 400, (c) 700 e (d) 800 passos de tempo.

Para a construção desses *clusters Beowulf* foi idealizado um esquema de segurança dos dados que segue o modelo recomendado por Pourzani *et al.*

[15]. Esse sistema foi adaptado de acordo com a usabilidade do sistema. Na Fig. 4.6 é apresentado o esquema de segurança implementado.

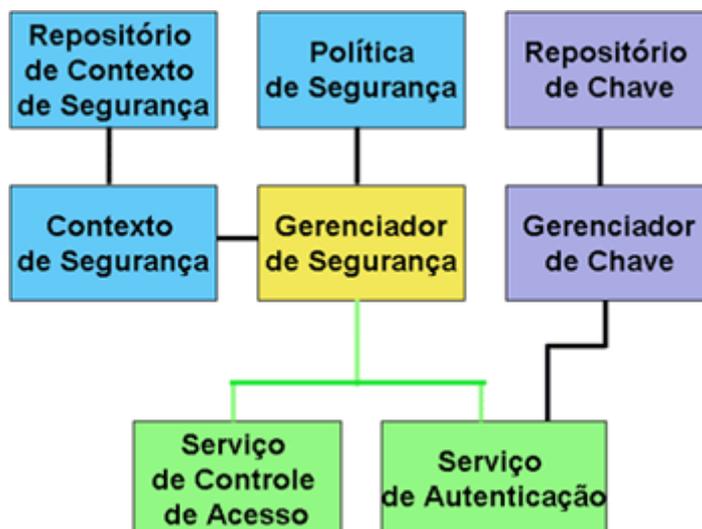


Fig 4.6 – Infra-estrutura de segurança adotado nos *clusters Beowulf*, seguindo o modelo de Pourzandi *et al.*[21].

O arranjo das máquinas apresentado na Fig 4.4 é interconectado através de um Hub 10/100 Mbps (ENH916P-NWMY-E) de 16 portas. Nessa figura é possível notar que existem computadores de diferentes arquiteturas, sendo conhecida como arquitetura heterogênea para processamento paralelo [54]. Além disso, é possível verificar que essa não é uma estrutura ideal, pois tem a máquina menos favorecida como servidor. Para analisar o desempenho individual de cada computador do *cluster*, foi realizado um teste variando o domínio computacional dos campos eletromagnéticos para simular a cavidade ressonante. A Tabela 4.2 traz os tempos despendidos pelas simulações de três mil passos de tempo, com resultados apresentados em segundos.

Tabela 4.2 – Tempo despendido em segundos para simulações de cavidades ressonantes 2D.

Dimensões do campo (células)	AEL1	AEL2	AEL3	AEL5	AEL8	AEL10
1000x1000	413	494	464	1702	535	538
2000x2000	1658	1755	1901	6825	2033	2114
3000x3000	3301	3427	3729	14443	4769	4155

Pela Tabela 4.2 é possível notar que a máquina AEL5 é bastante lenta quando comparada com as outras envolvidas no *cluster*. As máquinas AEL1 e AEL3 são as mais eficientes no processamento do domínio computacional 1000x1000, porém quando o domínio aumenta para 3000x3000, a AEL3 perde

seu posto qualitativo para a AEL2. As máquinas AEL1 e AEL3 possuem dois processadores Intel Xeon, que é uma estrutura considerada pelo fabricante como sendo mais moderno e eficiente que o Intel Pentium IV. Porém, nesses testes não foram aproveitados todos os recursos disponibilizados nesta arquitetura, sendo necessários estudos com a opção de *kernel* multiprocessamento simétrico (SMP), devido aos recursos de emulação de dois processadores em um. Essa afirmação comprova-se no sistema operacional Linux (Fedora) através do registro no arquivo `/proc/cpuinfo`.

Os recursos de dualidade dos processadores nas máquinas AEL1 e AEL3 precisam ser acionados de alguma maneira; neste caso, foi através da troca de mensagens entre os processadores, utilizando o *kernel* SMP para o aproveitamento de todos os recursos. Na Fig. 4.7, são mostrados os ganhos de desempenho obtidos nas máquinas AEL1 e, na Fig. 4.8, os ganhos na AEL3, aproveitando os recursos computacionais através da opção de *kernel* SMP.

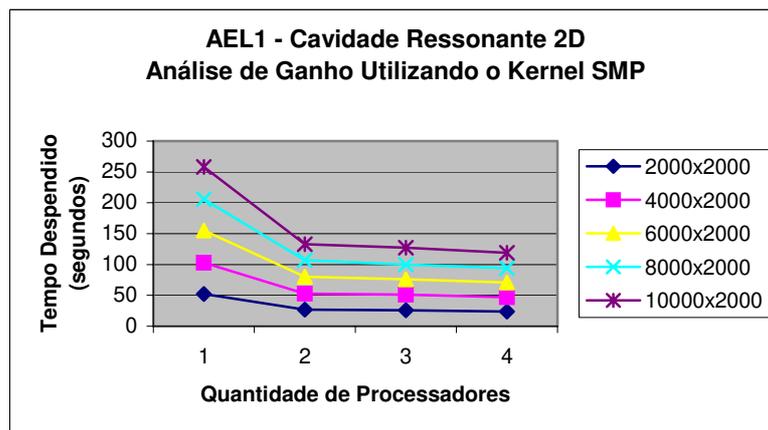


Fig. 4.7 – Gráfico do desempenho dos recursos do AEL1, com 100 passos de tempo.

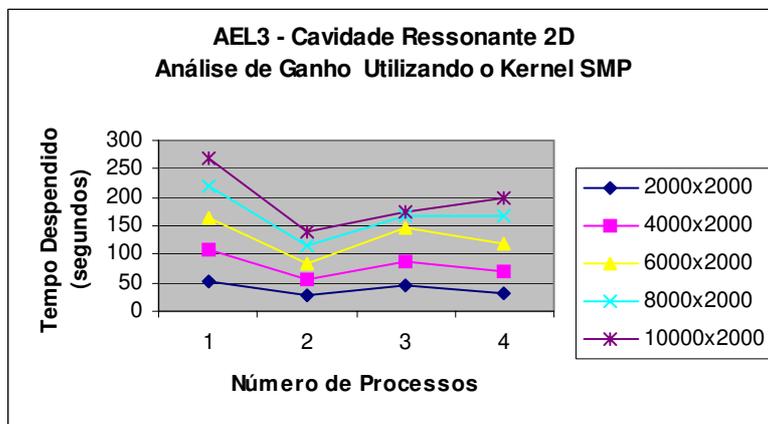


Fig. 4.8 – Gráfico de desempenho dos recursos do AEL3, com 100 passos de tempo.

Em decorrência dos resultados obtidos na Tabela 4.2, que mostra o desempenho individual dos equipamentos, foi refeita a configuração do *cluster Beowulf* para testar quais seriam os possíveis ganhos com essa nova estrutura. Na Fig. 4.9 é apresentado o novo arranjo do *cluster*, o qual não utiliza a máquina AEL5 na interconexão, que constituía um gargalo da estrutura, como comprovado na Tabela 4.2., pois o tempo de solução de um mesmo problema demanda muito mais tempo que o despendido em outras máquinas.

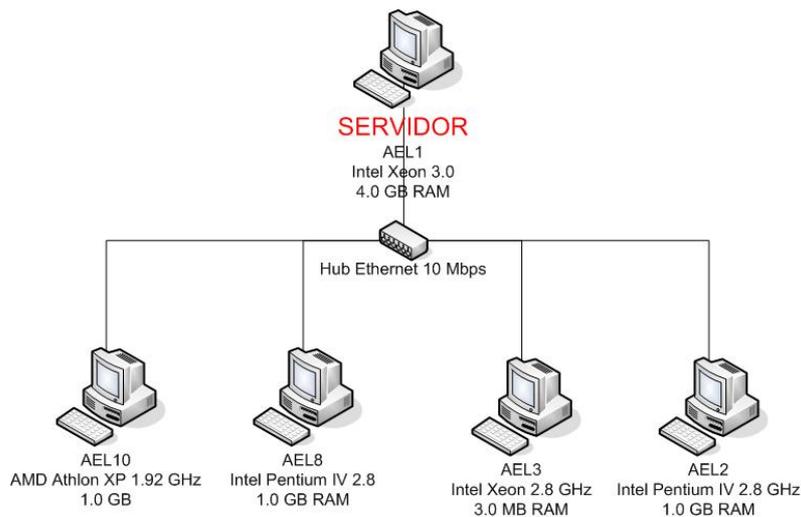


Fig. 4.9 – Configuração atual do cluster Beowulf do AEL, (CLUSTER-AEL-2).

Os testes realizados nessa configuração do *cluster Beowulf* possuem domínios computacionais maiores, devido à arquitetura ser mais eficiente que a anterior, justamente por utilizar o computador mais sofisticado como servidor. Na Fig. 4.10 é apresentado o gráfico com os resultados das simulações de uma cavidade ressonante de duas dimensões utilizando as máquinas AEL1, AEL2 e AEL3.

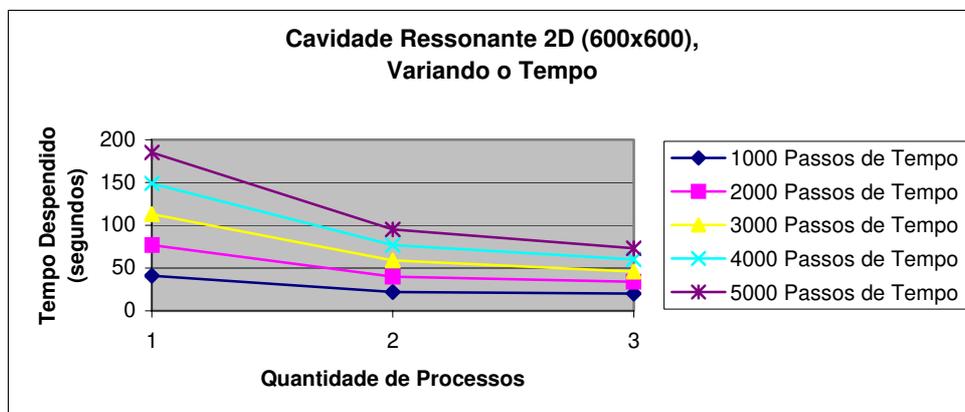


Fig. 4.10 – Tempo despendido na simulação 2D variando os passos de tempo.

Nessas simulações o segundo domínio computacional (600x600 células) é maior que o apresentado no gráfico da Fig. 4.3 (200x200 células); traçando um comparativo entre esses dois testes, com a mesma quantidade de passos de tempo, é notável como a estrutura do *CLUSTER-AEL-2* é mais eficiente que a do *CLUSTER-AEL-1*. Essa afirmação baseia-se no comparativo que relaciona os domínios computacionais com 360.000 e 40.000 itens, que foram calculados através dos métodos FDTD 2D, apresentado na Fig. 4.11, onde o gráfico apresenta os resultados. Pode ser visto que a relação de grandeza entre esses domínios é de nove vezes, porém o tempo gasto para solucionar o domínio computacional maior ficou apenas 50% acima do necessário para concluir a simulações do domínio menor, ou seja, apenas 1,5 vezes mais demorado.

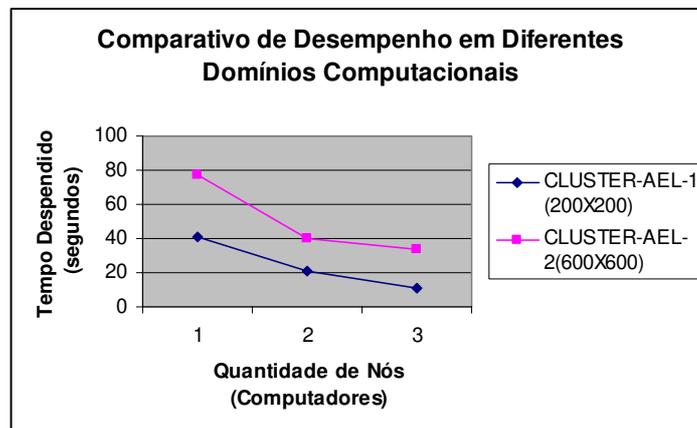


Fig. 4.11 – Comparativo de desempenho entre as estruturas de cluster

Outro importante resultado apresentado no gráfico da Fig. 4.10 é a constatação de valores que estão de acordo com a Lei de Amdahl (ver Anexo C), que se utiliza para medições de desempenho em aplicativos paralelos, especialmente em *clusters*. Dessa forma, tem-se uma estrutura heterogênea condizente com as teorias da computação paralela, que apresenta bons resultados de desempenho em simulações de médio porte. Na Fig. 4.12 são apresentados os resultados dos valores de *speedup*, que são referentes ao quanto os algoritmos paralelos são mais rápidos que os seqüenciais. A Fig. 4.13 disponibiliza graficamente o quão eficiente é o método e, na Fig. 4.14, apresenta-se os resultados a partir da Lei de Amdahl da máxima eficiência do paralelismo.

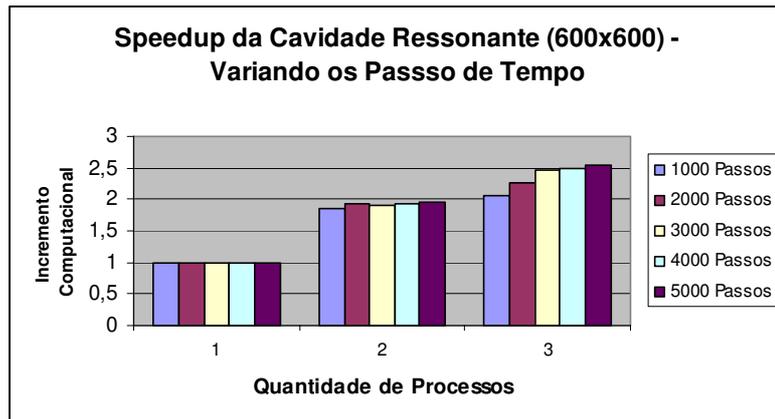


Fig. 4.12 – Comparando a rapidez para solucionar o FDTD 2D em Paralelo

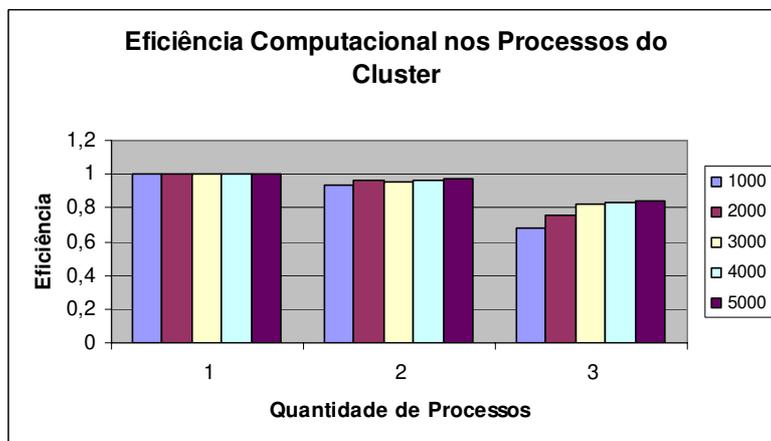


Fig. 4.13 – Eficiência computacional do processamento paralelo

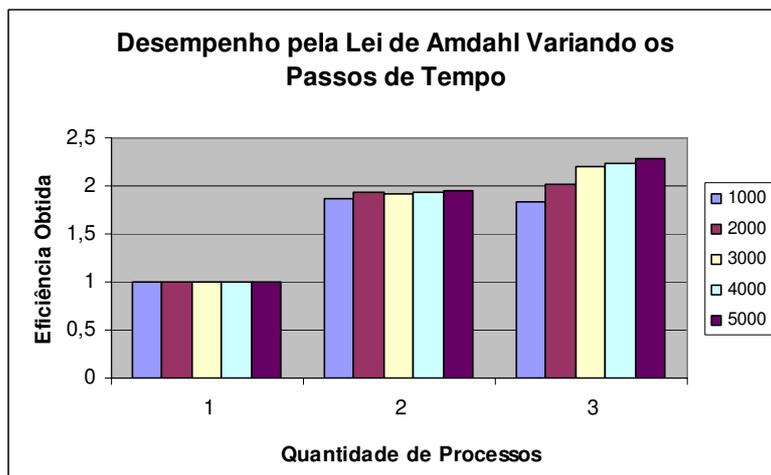


Fig. 4.14 – Cálculo da eficiência pela Lei de Amdahl

Nota-se, pelos três gráficos acima, que nessa estrutura de *cluster Beowulf* é mais interessante solucionar problemas mais complexos de serem analisados, neste caso, com mais passos de tempo. As simulações realizadas

nesse trabalho não são bastante dispendiosas em tempo, pois os computadores utilizados são os de uso geral do laboratório, o que inviabilizaria vários trabalhos de outros usuários. Porém esses testes de desempenho contribuem com a construção do conhecimento e o estabelecimento de certo parâmetros interessantes a serem considerados durante a paralelização dos métodos numéricos e o uso de uma estrutura de *cluster Beowulf* heterogênea para obter a computação de alto desempenho. É importante destacar, que os resultados apresentados até aqui mostraram que, efetivamente, o *cluster* é uma solução viável e interessante, por possibilitar que problemas numéricos de médio e grande porte sejam resolvidos.

Essa estrutura de paralelização pode apresentar resultados melhores quando aplicada em simulações de problemas eletromagnéticos tridimensionais, pois possibilita obter resultados teóricos mais precisos, dependendo da aplicação. No caso do método FDTD 3D, existem inúmeras simulações que são mais interessantes de serem analisadas tridimensionalmente como, por exemplo, em análises de efeitos dos celulares na cabeça humana e no estudo dos campos eletromagnéticos em estruturas com metamateriais. Nesses casos, a demanda computacional pode ser grande, tornando-se conveniente a paralelização do método e o uso de *clusters Beowulf* para agilizar o trabalho. No gráfico da Fig. 4.15 é mostrado o *speedup* nas simulações de uma cavidade ressonante tridimensional, em que fixou-se em duzentos passos de tempo e foram feitas modificações nas dimensões do domínio computacional cúbico.

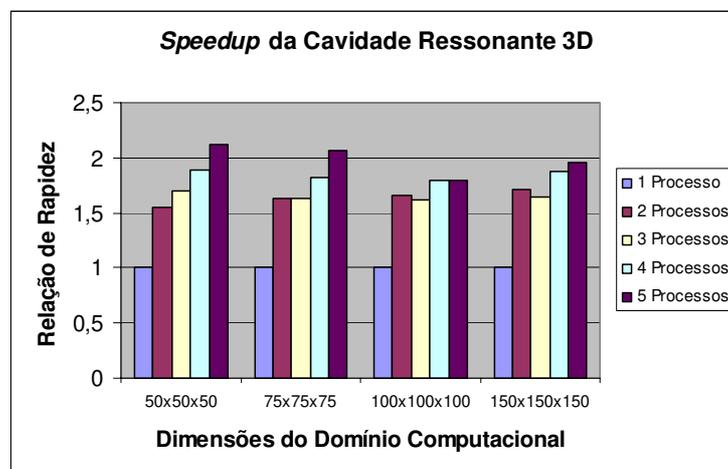


Fig. 4.15 – Speedup obtido na cavidade ressonante 3D

Com essas informações nota-se que o desempenho da paralelização se reduz quando são inseridos outros processos, mesmo tendo diminuído o tempo necessário para realizar as simulações. Esse resultado é satisfatório, porém não é muito bom, quando visto sob o prisma da demanda de melhoria no desempenho da solução a partir da paralelização do método FDTD. As simulações acima serviram para analisar domínios computacionais com diferentes dimensões e com os mesmos passos de tempo, sendo interessante avaliar resultados em que o domínio é fixado e ocorrem variações na quantidade de passos de tempo. Isto é apresentado na Fig. 4.16, com relação à eficiência da paralelização, em que são amostrados os dados da paralelização do FDTD-3D para um campo eletromagnético computacional cúbico de 100 células (100x100x100).

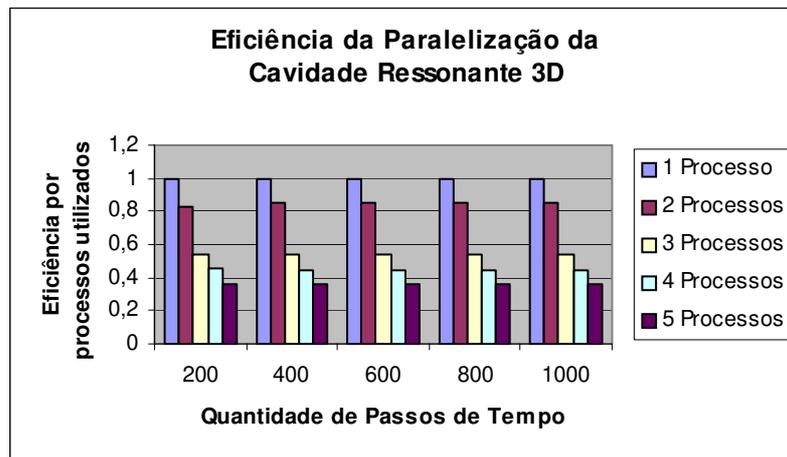


Fig. 4.16 – Eficiência da paralelização do FDTD 3D (100x100x100)

A eficiência apresentada acima é um comparativo entre o tempo despendido em uma simulação seqüencial (em um processador) dividido pelo tempo em paralelo e estando de acordo com a quantidade de processadores do agregado. Para exemplificar, o tempo de processamento de um cálculo em uma máquina é de 50 segundos, teoricamente, espera-se que a divisão do problema em 2 máquinas faria com que o tempo fosse de 25, porém existem fatores limitantes na comunicação dos processos que inviabilizam que isso ocorra, fazendo com que haja redução do desempenho, imaginando que o tempo final seja de 30 segundos. Desta forma, tem-se que a eficiência foi de 0,83333, ou seja, o ganho computacional de acordo com a eficiência é de 83% em relação ao 100% teórico.

Nas simulações do FDTD 3D nota-se que houve uma redução bastante acentuada no desempenho, especialmente quando o computador AEL3 é inserido no *cluster*, mesmo este sendo um equipamento bastante robusto. Os dois principais motivos para essa ocorrência pode ser o fluxo de dados que trafegam na rede e/ou o não aproveitamento dos recursos computacionais possivelmente disponibilizados pelos equipamentos. O fluxo, nesse caso, pode ser desconsiderado, pois a seguir o tempo de processamento é reduzido com a inserção da máquina AEL10. Por esse motivo, notou-se que a forma de paralelização não era eficiente o bastante, havendo a necessidade de otimizar os códigos para que eles possam utilizar melhor os recursos disponíveis. Essa afirmação também se baseia em outros testes que variaram o dimensionamento dos domínios e percebeu-se que havia a necessidade de melhorias, pois a capacidade de processamento que estava bastante comprometida, inviabilizando a solução de problemas de grande porte.

Por esse motivo, foram feitas otimizações no método das diferenças finitas no domínio do tempo em duas dimensões, para que fossem mais adequados às condições de paralelização. Os algoritmos foram construídos para a distribuição dos cálculos e o parcelamento da alocação de memória entre os nós (máquinas) do *cluster*. Com isso, a solução tornou-se mais eficiente e mais próxima de uma implementação computacionalmente boa, pois ainda existem muitas formas de melhoramento dos algoritmos. No algoritmo de FDTD-3D as otimizações estão por serem realizadas.

A Fig. 4.17 mostra um gráfico com os resultados da eficiência obtida com esse novo algoritmo otimizado para simulações bidimensionais utilizando o método FDTD, nesse caso, para analisar cavidades ressonantes 2D.

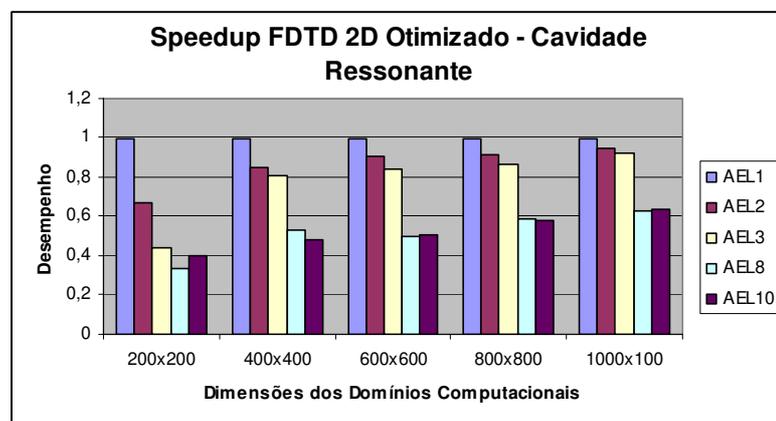


Fig. 4.17 – Eficiência do paralelismo do FDTD 2D melhorado

No gráfico da Fig. 4.17 é possível verificar que a eficiência do paralelismo melhorou, decorrência do melhor aproveitamento dos recursos de processamento e da memória dos computadores. Confirma-se o ganho de desempenho do algoritmo paralelizado do método FDTD-2D para simular uma cavidade ressonante. Continuando a analisar do gráfico da Fig. 4.17, nota-se que a eficiência do paralelismo é mais interessante nas estruturas maiores como, por exemplo, a de dimensões 1000x1000, sendo importante avaliar esses resultados segundo a Lei de Amdahl, para comprovar, de outra forma, esse bom desempenho do paralelismo. A Fig. 4.18 traz um gráfico com a eficiência do paralelismo do método FDTD-2D usando a Lei de Amdahl.

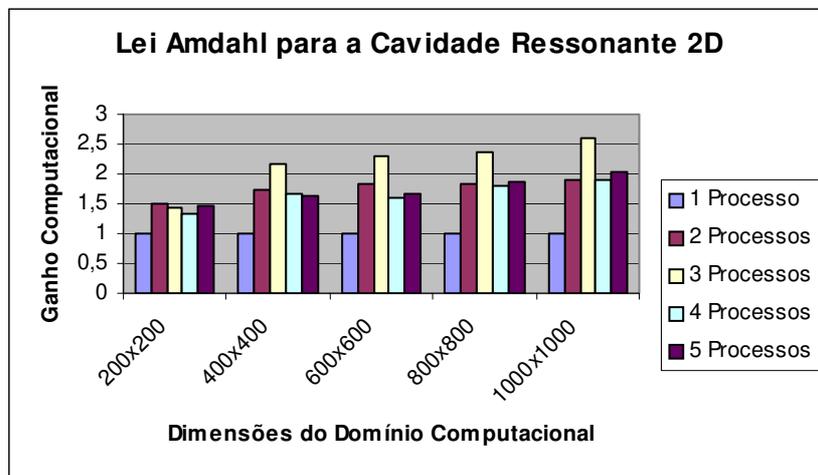


Fig. 4.18 – Análise do desempenho do paralelismo do FDTD-2D pela Lei de Amdahl

Com a estrutura deste *cluster Beowulf* (CLUSTER-AEL-1) e o desenvolvimento mais adequado dos algoritmos paralelos, nota-se que os tempos despendidos nas simulações são bastante reduzidos. Essa evolução é condizente com o conhecimento construído e a experiência adquirida com as simulações e desenvolvimento dos códigos. Nos gráficos da Fig. 4.19 são mostrados os dados sobre o desempenho das simulações de estruturas bidimensionais de médio e grande porte, para uma cavidade ressonante em que se varia as dimensões do domínio computacional e se fixa a quantidade de passos de tempo em mil intervalos. Para esse teste, resolveu-se alterar a seqüência de inserção das máquinas no *cluster Beowulf*, valendo-se agora da seguinte: AEL1, AEL2, AEL3, AEL10 e AEL8.

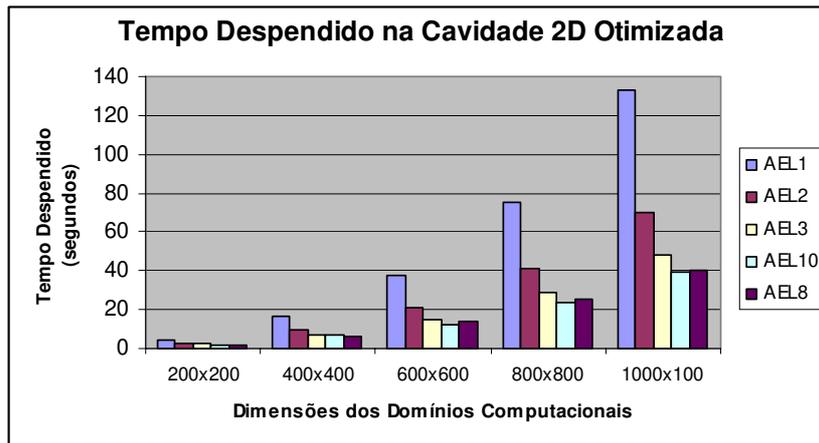


Fig. 4.19 – Tempo gasto nas simulações de médio e grande porte da cavityde ressonante 2D

Essa simulação é muito importante para se ter uma noção do ganho real de tempo que é possível ter com a computação massivamente paralela em problemas de médio e grande porte, visto que as dimensões para os campos eletromagnéticos da cavityde ressonante são razoavelmente grandes. Pelo gráfico é possível perceber que o tempo de espera das simulações são reduzidos como, por exemplo, no caso mais complexo que é o domínio com seis mil linhas e seis mil colunas, que gera uma matriz com um total de 36.000.000 de itens. O tempo demorado para o processamento numa máquina é de 4139 segundos, que é cerca de uma hora e oito minutos de simulação. Quando é executado o processamento paralelo em duas máquinas, esse tempo despenca para 2103 segundos, o equivalente a 35 minutos e quando são utilizadas todas as cinco máquinas do *cluster* o tempo cai para 21 minutos. O *speedup* dessa solução massivamente paralela é apresentado na Fig. 4.20.

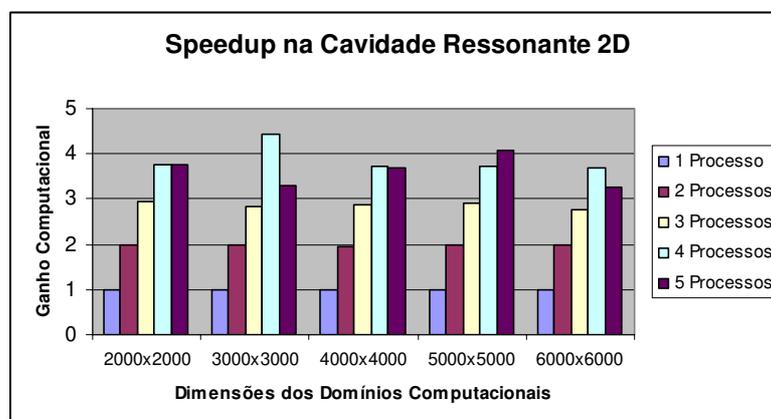


Fig. 4.20 – Speedup da cavityde ressonante 2D

4.2.3 FDTD Paralelo para Análise da SAR na Cabeça Humana

Nas últimas duas décadas o eletromagnetismo computacional vem recebendo crucial importância no desenvolvimento de várias ciências e tecnologias, tais como biologia, telecomunicações, fotônica e muitas outras [55], [40], [56].

Recentemente, pesquisadores em todo o mundo têm estudado os efeitos da penetração das ondas eletromagnéticas no corpo humano, especialmente na região da cabeça [56]. O principal foco dos trabalhos vem sendo o uso dos telefones celulares, que emitem sinais eletromagnéticos de 800 MHz a 1100 MHz. A absorção dessas ondas tem sido o foco dos estudos dos efeitos fisiológicos térmicos e não-térmicos.

Essas simulações podem demandar grande poder computacional, como memória e processador. Para isso, seriam necessários supercomputadores, ou algoritmos especiais para solucionar esses problemas complexos.

Para satisfazer essa demanda por recursos computacionais, neste trabalho foi proposto o emprego do *cluster Beowulf* heterogêneo para o processamento massivamente paralelo para o método FDTD, utilizando a biblioteca LAM-MPI para a troca de mensagens. Essa solução deu bons resultados de desempenho, viabilizando a exploração de qualquer parque computacional, independente da plataforma. Além disso, os acréscimos de recursos computacionais não acarretaram qualquer gasto financeiro extra.

4.2.3.1 Método SAR para a Cabeça Humana

A representação computacional da cabeça humana para análises da propagação de microondas eletromagnéticas é uma tarefa complexa de ser realizada, pois possui diversos tecidos com propriedades elétricas diferentes e densidades de massa.

Utilizando o sistema de Imagens de Ressonância Magnética (MRI), o modelo da cabeça humana foi dividida em 54 cortes transversais (camadas) de 5 mm de espessura, que foram discretizadas para compor um modelo da cabeça humana que é cúbico, com 40x40 pixels, para análises computacionais, como na Fig. 4.21 [31].

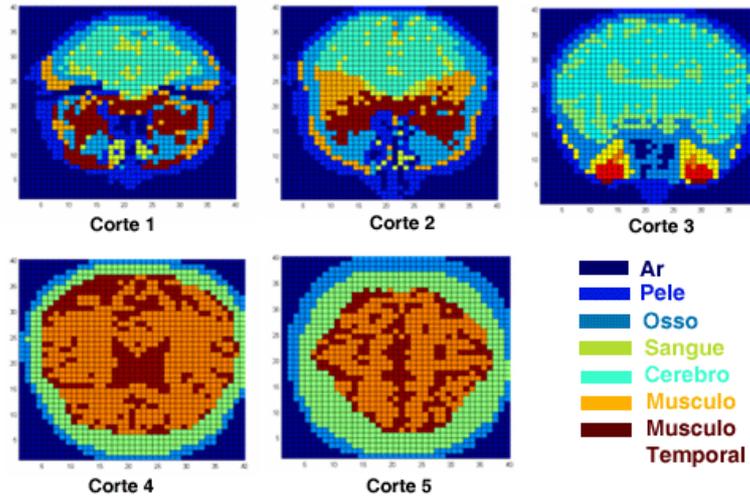


Fig. 4.21–Cortes transversais (12, 15, 25, 35 e 45) com os tecidos mais importantes.

O parâmetro dosimétrico a ser analisado é a Taxa Específica de Absorção (SAR). A SAR gera efeitos térmicos, e pode ser definida como sendo a taxa de aumento de energia absorvida $[\partial W]$ por elemento de massa $[\partial m]$ contido em um volume $[\partial V]$ do elemento, com uma massa específica ρ [52]. Pode-se escrever:

$$SAR = \frac{\partial}{\partial t} \frac{\partial W}{\partial m} = \frac{\partial}{\partial t} \frac{\partial W}{\partial(\partial V)} \quad [mW / g] \quad (4.1)$$

Alternativamente, a SAR pode ser expressa em termos dos campos elétricos, aplicando o Teorema do Vetor de *Poyting* [25] para campos eletromagnéticos com excitação senoidal no domínio da frequência, pela seguinte equação:

$$SAR = \frac{\sigma}{2\rho} |E_i|^2 = \frac{|J_i|^2}{2\rho\sigma} \quad [mW / g] \quad (4.2)$$

onde E_i , é o campo elétrico e J_i , é a densidade de corrente em um ponto.

Para o domínio do tempo, é calculada a média local da SAR em termos do tempo total da simulação (N_{\max}).

$$SAR = \frac{\sigma}{2\rho} \frac{1}{N_{\max}} \sum [E_x^2 + E_y^2] \quad [mW / g] \quad (4.3)$$

Essa aplicação foi implementada para solucionar, em paralelo, o modo TE_z [28], para analisar e medir o desempenho do *cluster Beowulf* heterogêneo e resolver campos eletromagnéticos utilizando o método FDTD-2D e PML,

visando o desenvolvimento de uma aplicativo para simulações complexas como no caso de estudos de SAR (Fig. 4.22). A primeira configuração do *cluster Beowulf* utilizado nos testes é o CLUSTER-AEL-1.

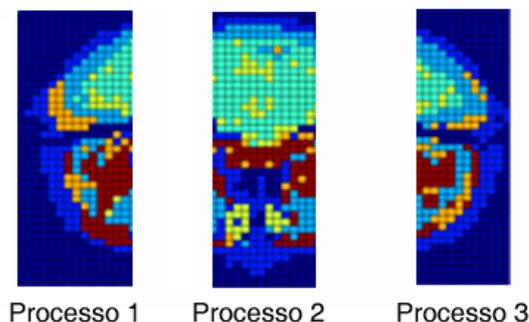


Fig. 4.22 – Corte 1 dividido entre 3 processos

Na Fig. 4.23 são apresentados os resultados do tempo despendido para simular a SAR na cabeça humana, lembrando que este teste foi realizado em uma estrutura, já mencionada, como não sendo a ideal, pois o servidor é o computador menos favorecido da rede.

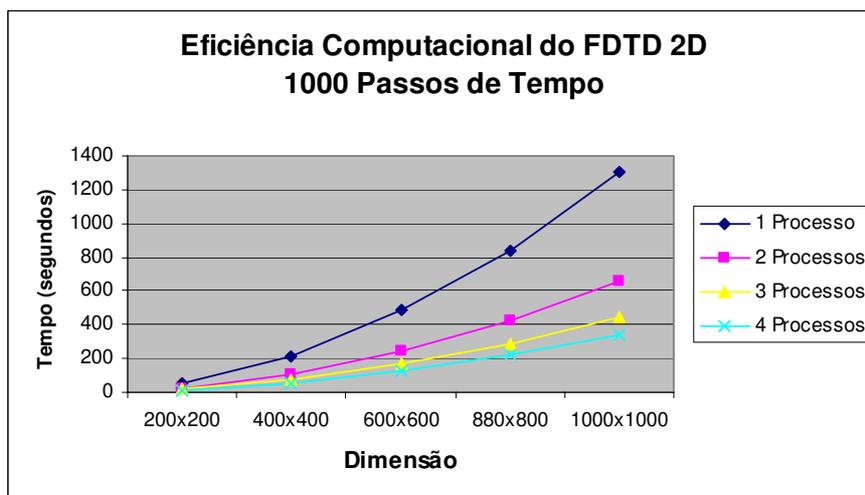


Fig. 4.23 – Tempo gasto para as simulações da SAR na cabeça humana

Os resultados mostrados na Fig. 4.23 demonstram que o ganho computacional foi bom, pois o tempo de resposta reduziu bastante. Apesar de não ser a solução ideal, percebe-se que essa seria uma boa forma de incrementar o desempenho dos recursos computacionais já existentes, principalmente pelo baixo custo envolvido no aproveitamento do parque computacional do laboratório e pelo uso de *software* livre, com base no sistema operacional Linux, da distribuição Fedora Core 2 como a biblioteca para a troca

de mensagens LAM-MPI. A eficiência dessa solução extrapola a Lei de Amdahl.

Essas simulações utilizaram 0,98 GHz de frequência e células cúbicas de 0,0025 metros para o incremento espacial. A fonte elétrica foi um dipolo[em E_y] e as propriedades dos materiais são dadas na Tabela 4.3 [31].

Tabela 4.3 – Índice de propriedades dos materiais da cabeça humana

Materiais	ϵ_0	δ	ρ
Ar	1,0	0,0	12e-4
Pele	41,41	0,87	1,0
Osso	5,5	0,04	1,2
Cérebro	51,0	1,23	1,05
Sangue	61,36	1,54	1,0
Músculo	55,96	0,97	1,04
Músculo Temporal	69,90	1,64	1,01

Para facilitar o entendimento sobre a configuração da estrutura do problema, a Fig. 4.24 mostra a evolução dos cálculos do FDTD no campo E_y no trigésimo passo de tempo, para expor os resultados da fonte e do campo nesse instante de tempo.

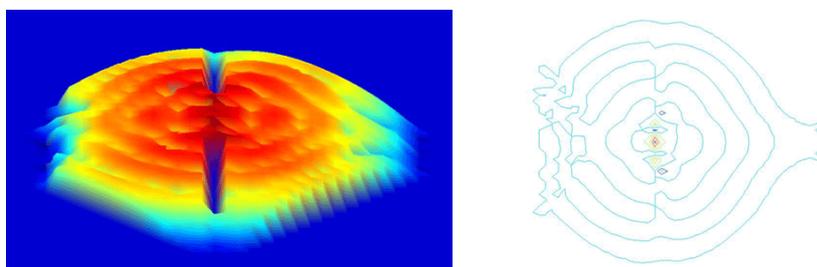


Fig. 4.24 – Fonte e campo E_y no trigésimo passo de tempo da Simulação

O resultado final da simulação da SAR é um campo simétrico de 200 células e analisado após 1000 passos de tempo, como mostrado na Fig. 4.25. Essa figura mostra as cinco camadas da cabeça e seus respectivos resultados SAR em valores absolutos da escala logarítmica, sendo possível perceber a presença de ondas superficiais na cabeça humana.

Nessa figura é possível ver que concentração mais significativa de SAR na cabeça humana ocorre no lado em que se encontra a fonte (antena). Isso comprova a importância de testes precisos dos efeitos eletromagnéticos na cabeça humana. Com isso, percebe-se a importância da construção de um

modelo mais preciso da cabeça humana, pois provavelmente serão obtidos resultados diferentes desse trabalho em que a discretização da cabeça humana é bem simplificada.

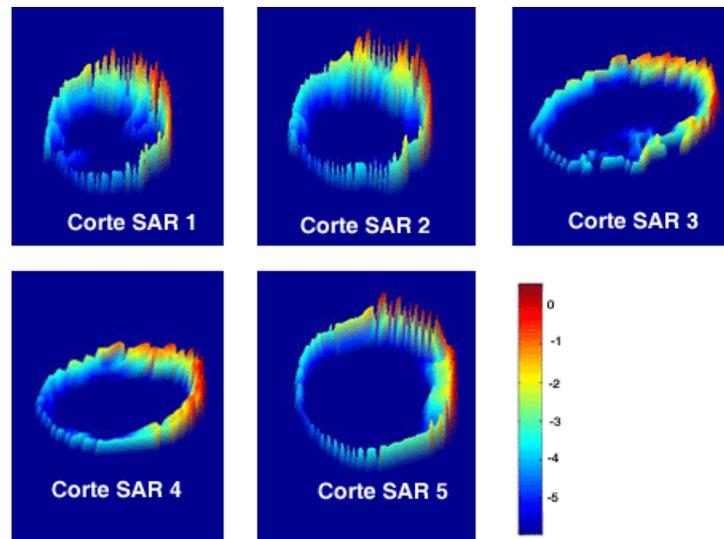


Fig. 4.25 – Resultado da SAR na cabeça humana após 1000 passos de tempo

Um outro teste para a simulação da SAR foi realizado na configuração mais eficiente do *cluster Beowulf*, ou seja, no CLUSTER-AEL-2, onde o novo servidor é o melhor computador da rede. Nessa estrutura ocorreu o mesmo resultado já obtido num comparativo entre as diferentes estruturas de *cluster*: o tempo da solução foi reduzido. Como mostrado na Fig. 4.26 e na Fig. 4.27, o tempo de processamento reduz-se bastante, porém a eficiência do paralelismo relacionando o número de processadores decaiu acentuadamente, resultado provável da limitação da estrutura da rede de computadores disponível.

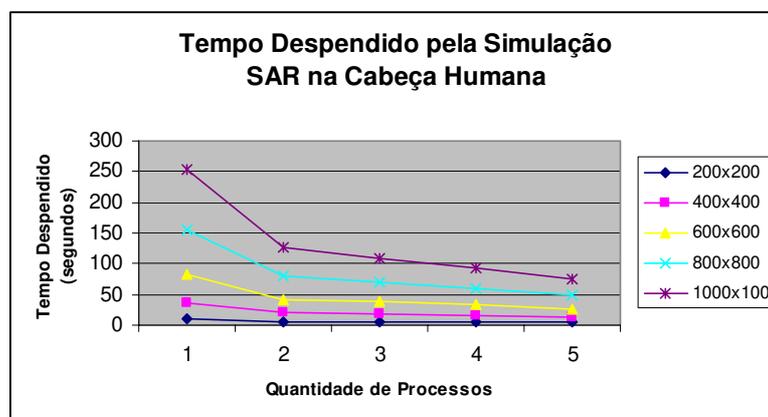


Fig. 4.26 – Tempo despendido na simulação a SAR 2D da cabeça humana

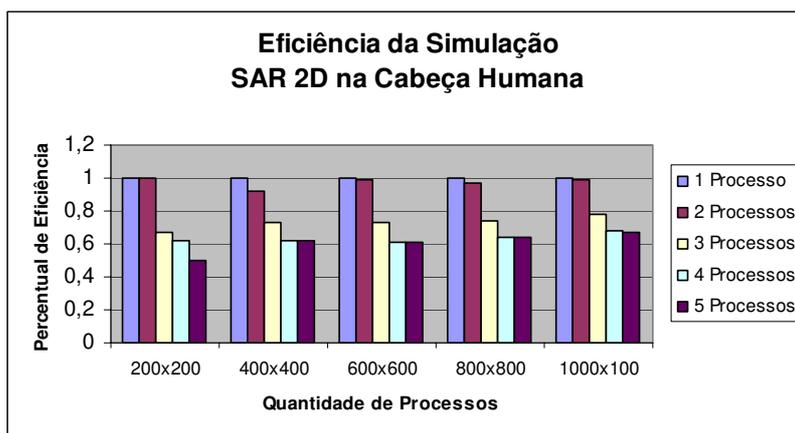


Fig. 4.27 – Eficiência da paralelização da simulação 2D da SAR na cabeça humana

Outra simulação realizada para analisar o desempenho da computação paralela em estudos de SAR na cabeça humana é mostrada a seguir, na Fig. 4.28, onde é apresentada a eficiência do paralelismo num domínio computacional de 400x400, variando os passos de tempo de 1000 a 4000 intervalos de tempo.

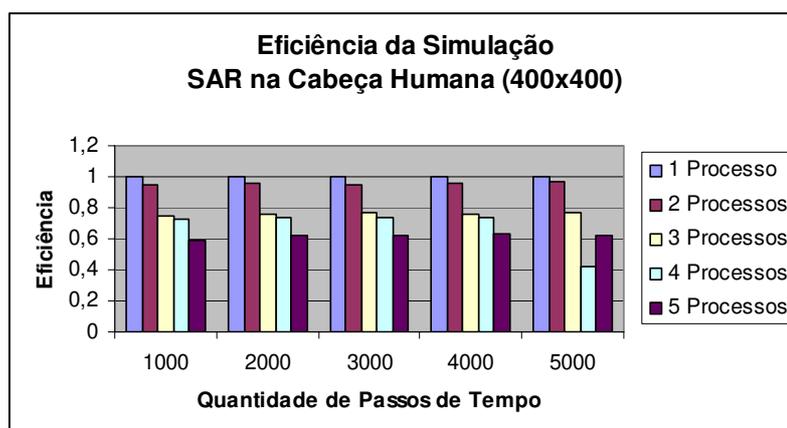


Fig. 4.28 – Eficiência da Paralelização da SAR 2D (400x400) variando os passos de tempo

Com esses testes possibilita concluir que o emprego de *cluster Beowulf* nas simulações de bidimensionais da SAR na cabeça humana é uma boa solução para os problemas de desempenho e recursos computacionais dispendiosos, valendo ressaltar novamente a redução nos custos financeiros para o incremento computacional e a possibilidade de melhoria na atual estrutura de testes.

4.2.4 FDTD Aplicado aos Estudos de Metamateriais

Os metamateriais são materiais que possuem a permissividade elétrica e a permeabilidade magnética negativos tendo sido apresentado por Veselago [58] em 1968, que é conhecido como material da mão-esquerda (*left-handed material*) devido aos seus vetores de onda e de Poyting estarem em direções opostas [59].

Veselago [58] definiu que algumas regras do eletromagnetismo estão associadas essencialmente aos sinais da ε e μ . Essa relação está primeiramente descrita nas Equações de Maxwell e nas relações constitutivas.

$$\nabla \times E = -\frac{1}{c} \frac{\partial B}{\partial t} \quad (4.4)$$

$$\nabla \times H = \frac{1}{c} \frac{\partial D}{\partial t} \quad (4.5)$$

$$B = \mu H \quad (4.6)$$

$$D = \varepsilon E \quad (4.7)$$

Para uma onda plana monocromática, em que todas as quantidades são proporcionais a $e^{i(kz - \omega t)}$, as equações de 4.4 a 4.7 são reduzidas a

$$[kE] = \frac{\omega}{c} \mu H \quad (4.8)$$

$$[kH] = -\frac{\omega}{c} \varepsilon E \quad (4.9)$$

Nesse caso, quando o ε e μ são maiores que zero; então E , H , e k formam um conjunto vetorial da mão direita e, se $\varepsilon < 0$ e $\mu < 0$, eles são da mão esquerda. Quando são adicionados os co-senos para os vetores de E , H , e k e denotados em termos do meio, esses elementos serão caracterizados pela matriz G .

$$G = \begin{bmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \beta_1 & \beta_2 & \beta_3 \\ \gamma_1 & \gamma_2 & \gamma_3 \end{bmatrix} \quad (4.10)$$

O determinante dessa matriz é igual +1 se os vetores E , H , e k formam um conjunto da mão direita, e -1 se for um conjunto da mão esquerda. O determinante dessa matriz será indicado por p , que a seguir pode ser inserido

na Lei de Snell para calcular a reflexão e refração na interface entre meios [58], como mostrado na Fig. 4.29.

$$\frac{\text{sen } \varphi}{\text{sen } \psi} = n_{1,2} - \frac{p_2}{p_1} \sqrt{\frac{\varepsilon_2 \mu_2}{\varepsilon_1 \mu_1}} \quad (4.11)$$

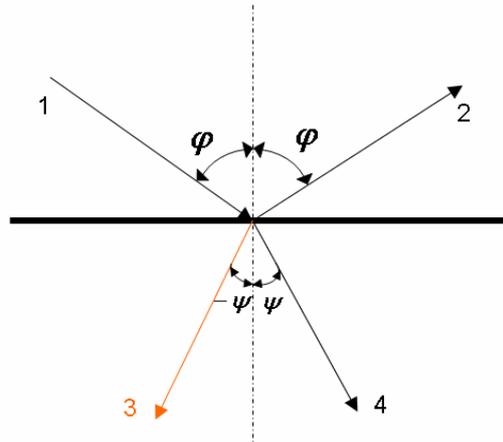


Fig. 4.29 – Lei de Snell com uma reflexão negativa

A experimentação e a teorização desse trabalho voltaram a ser estudado, em 1999, por Pendry *et al.* [60], quando eles propuseram algumas estruturas de cristais fotônicos para análises da banda proibidas (*band-gap*). Em 2001 foram iniciados os cálculos numéricos, via método FDTD, implementados por Ziolkowski e Hayman [61], para comprovar numericamente que algumas estruturas de materiais artificialmente construídas não possuem características físicas encontradas na natureza como, por exemplo, algumas estruturas de fotônica de banda proibida (PBG) e os meios duplamente negativos (DNG) [61].

Em 2003, no Departamento de Microonda e Óptica (DMO), da Unicamp, Sartori [43] apresentou os resultados de seus trabalhos experimentais utilizando tubos de PVC justapostos na mini-câmara anecóica construída por ele.

As simulações numéricas de estruturas metamateriais analisadas, demandam recursos computacionais de médio porte, devido à complexidade e dimensões dos campos; foi isso o que nos motivou a implementação em duas dimensões desse método em paralelo. Também são motivadores desse trabalho a inovação tecnológica e a importância atual dessa área. No gráfico da Fig. 4.30 é apresentado o tempo despendido nas simulações de domínios

computacionais com dimensões variadas, para estudar os acontecimentos físicos ocorridos estrutura metamaterial apresentada na Fig. 4.31.

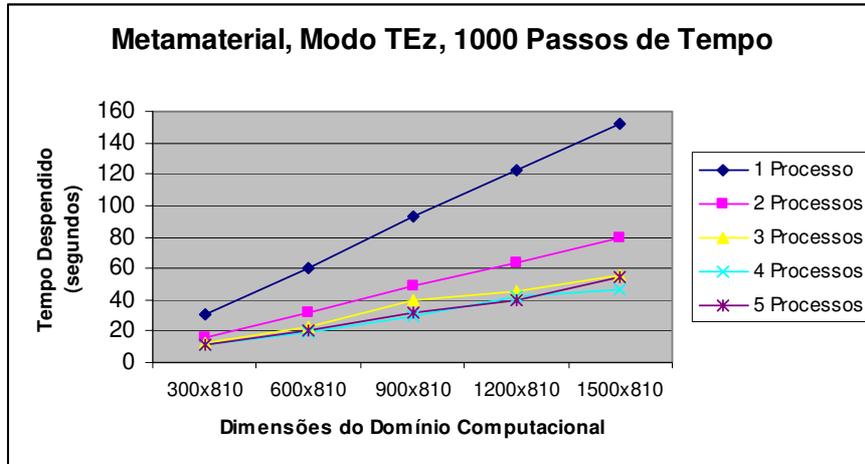


Fig. 4.30 – Desempenho da Simulação de Estruturas Metamateriais variando as dimensões

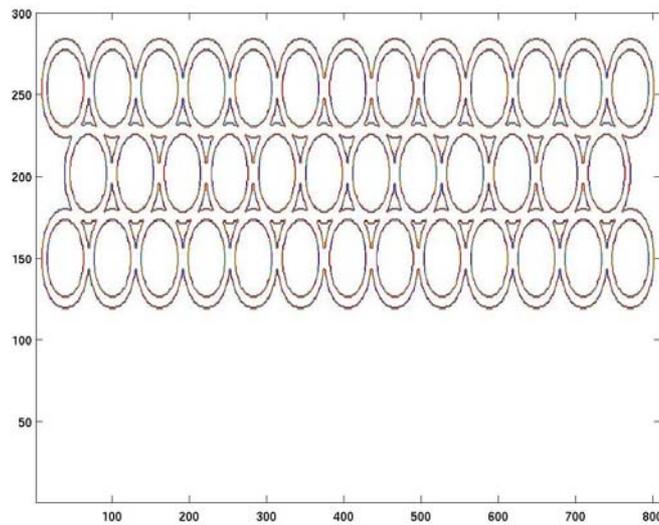


Fig. 4.31 – Estrutura metamaterial utilizada na simulação do campo 300x810

Nesse caso, nota-se o bom ganho de desempenho da solução em paralelo, fazendo com que o tempo das simulações seja bem reduzido. Nos gráficos da Fig. 4.32 é possível notar a eficiência da solução paralela em relação ao ótimo teórico da computação paralela.

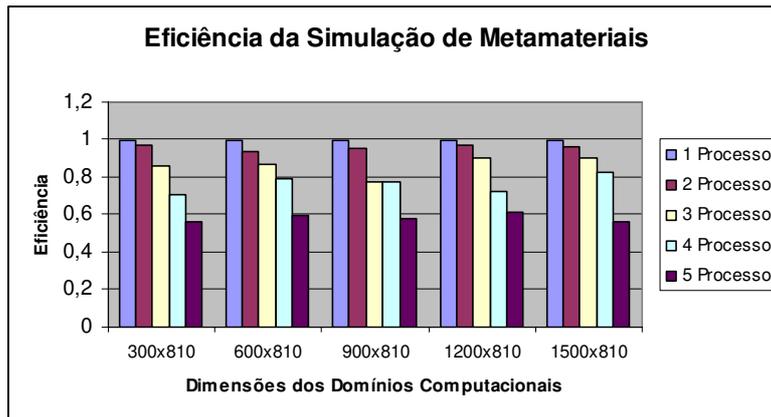


Fig. 4.32 – Análise de desempenho da paralelização do FDTD para estruturas metamateriais

Essa simulação foi realizada para a amostragem do desempenho do método paralelizado e, principalmente, para introduzir e analisar conceitos sobre as estruturas metamateriais, que são problemas complexos de serem estudados devido à grande demanda computacional. A estrutura metamaterial implementada para estudos físicos é apresentada na Fig. 4.33, tendo um dipolo como a fonte de excitação do campo E_y no modo TE_z paralelo que foi dividido, em processos, no eixo x .

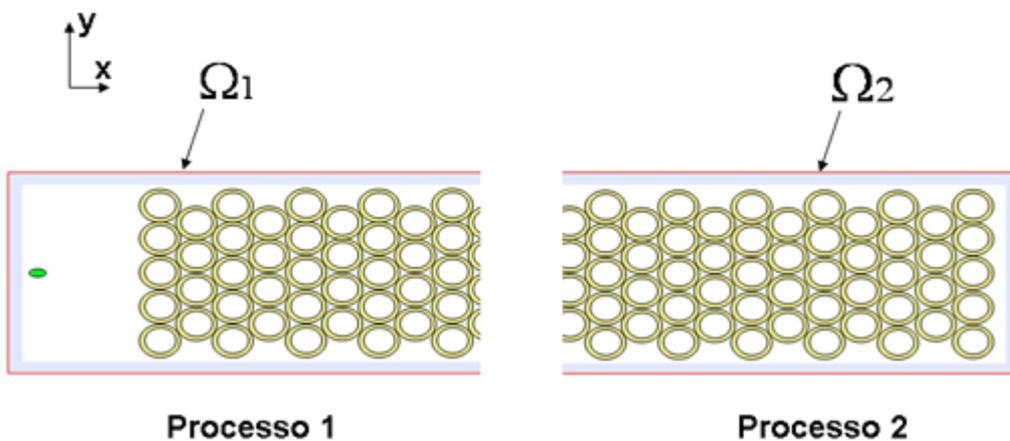


Fig. 4.33 – Estrutura metamaterial em paralelo para análises dos efeitos físicos

Na Fig. 4.34 é mostrado um gráfico que representa o desempenho obtido em simulações paralelas desse problema.

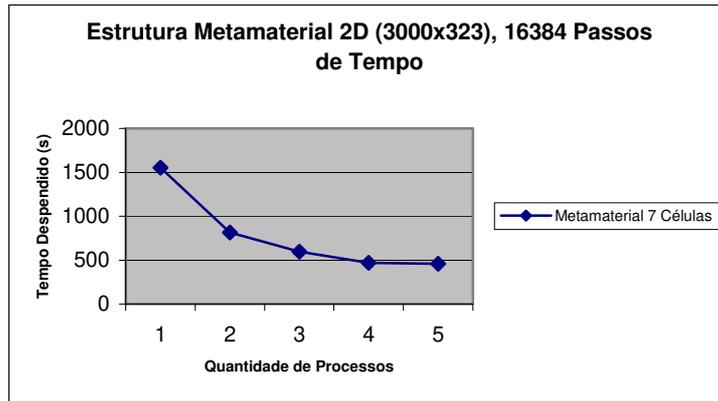


Fig. 4.34 - Tempo despendido para simular a estrutura metamaterial dividida em processos

De acordo com o gráfico acima, nota-se que o tempo gasto nas simulações pode ser bastante reduzido, sendo interessante analisar o *speedup* (Fig. 4.35). Assim, é possível comprovar, na prática, o bom desempenho dessa solução para o processamento massivamente paralelo.

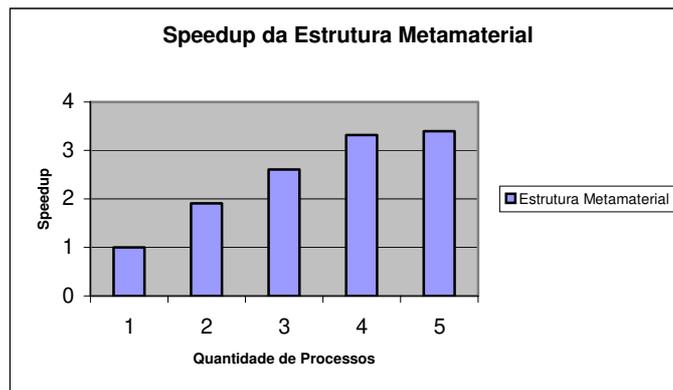


Fig. 4.35 – Speedup da simulação da estrutura metamaterial

Além dos testes de desempenho, foram realizadas algumas análises baseadas em cálculos obtidos na literatura [62] para a comprovação do método e dos resultados. Na Fig. 4.36 é apresentado um gráfico que mostra os valores das amplitudes das ondas, em diferentes freqüências, calculados discretamente de 7,8 GHz a 11,8 GHz, com um passo de 0,2 GHz.

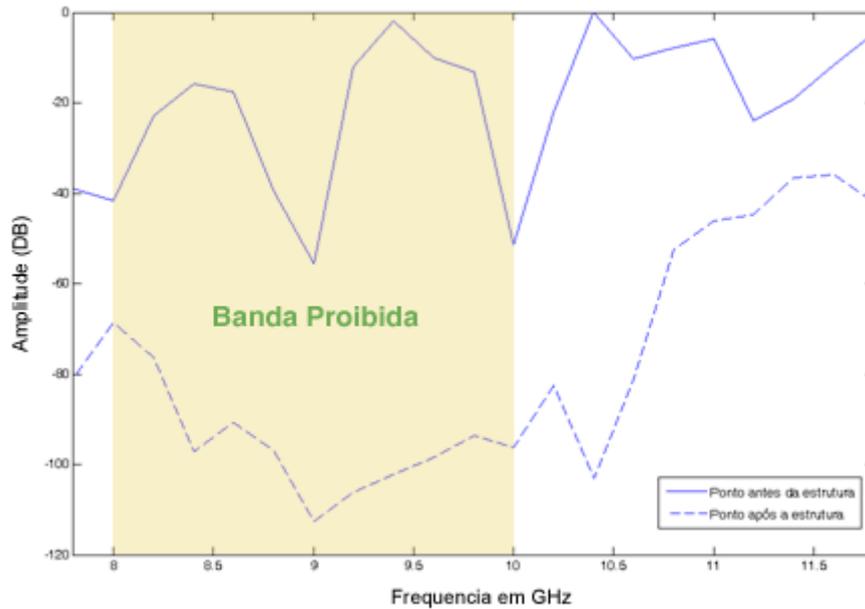


Fig. 4.36 – Banda Proibida da simulação da estrutura metamaterial, em dB

Para a comprovação numérica dos resultados acima, pode-se calcular a freqüência da banda proibida para a disposição hexagonal, que pode ser realizada da seguinte maneira:

$$f_{bg} = k \frac{c}{a}, \quad (4.12)$$

onde:

- F_{bg} Freqüência de banda proibida
- k Índice que define o intervalo de freqüência onde ocorre a banda proibida (~0,4 a 0,5)
- c Velocidade da luz no vácuo (3×10^8 m/s)
- a Distância entre os centros anéis da estrutura

O cálculo da banda proibida coincidiu com os resultados da simulação da estrutura. A distância entre os centros dos orifícios circulares é de 15 mm (Fig. 4.37) e utilizando os parâmetros que definem os intervalo de freqüência de 0,4 a 0,5, obteve-se as freqüências entre 8,0 GHz e 10,0 GHz, sendo equivalente ao apresentado no gráfico da Fig. 4.36.

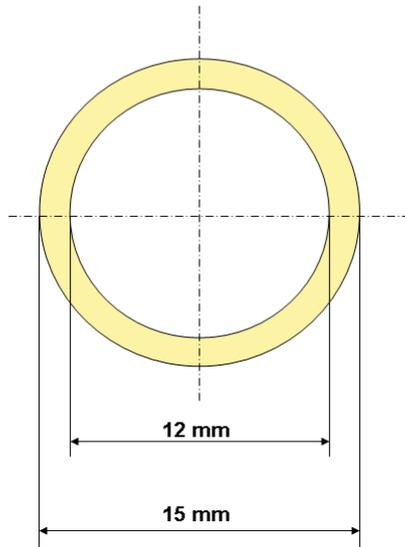


Fig. 4.37 – Anel base da estrutura metamaterial

Capítulo 5

Conclusões e Trabalhos Futuros

5.1 Conclusões

O desenvolvimento dos algoritmos numéricos paralelos conduz a tarefas mais complexas, porém mais eficientes. Esses códigos podem ser portáteis e independentes da estrutura, dos sistemas operacionais e, em alguns casos, de linguagens de programação, como ocorre com os aplicativos desenvolvidos no padrão MPI e que podem ser escritos em C, C++ e Fortran e podem se comunicar entre si. Para a produção desses programas é preciso ter experiência e conhecimento aprofundado, fatores requeridos pelas possibilidades de produção adequada dos algoritmos paralelos.

Nesse trabalho foram obtidos bons resultados de desempenho no processamento massivamente paralelo utilizando o *cluster* Beowulf para a execução das tarefas. Os testes utilizando o método FDTD em duas e três dimensões foram positivos nas simulações de estruturas de médio porte para as simulações das cavidades ressonantes que foram tomadas como base para as experiências, devido à eficiência dos algoritmos desenvolvidos.

Com esses resultados comprovou-se que o cluster Beowulf é um recurso importante para solucionar a demanda por recursos computacionais de alto desempenho, a um baixo custo financeiro, possibilitando a democratização de trabalhos complexos que demandam grande poder de processamento, pois a aquisição de supercomputadores é inviável para a maioria das instituições públicas e privadas, devido aos elevados preços desses equipamentos.

Especificamente nesse trabalho, o ganho de desempenho computacional não necessitou que fossem realizados investimentos em equipamentos e pacotes de programas, em decorrência do aproveitamento do parque computacional dos laboratórios do DMO e a utilização de aplicativos gratuitos e abertos. Os experimentos comprovam de maneira prática, na universidade, a possibilidade de implementar as propostas incentivadoras do Governo Federal em relação a adoção do *software* livre.

O pacote de programas está focado apenas ao provimento de recursos para solucionar problemas de médio e grande porte para o processamento massivamente paralelo, sendo bastante interessante estender ele a todas atividades, pois assim os gastos com *softwares* serão reduzidos, possibilitando que verbas que seriam destinadas para esses fins sejam aproveitadas de outras maneiras. Para o desenvolvimento optou-se pelo sistema operacional Linux, na distribuição Fedora (Core 2 e 3) por ser gratuito e de fácil uso, tendo a biblioteca LAM-MPI para troca das mensagens entre os processos e o compilador *gcc*. Todos esses *softwares* são padronizados nas normas GPL.

A comprovação desses resultados foi vista nos testes de aplicações de eletromagnetismo computacional, para analisar e estudar a SAR na cabeça humana e nas estruturas metamateriais. A realização dessas simulações nessas estruturas justifica-se pela quantidade de pesquisas que essas áreas têm demandado e a importância atual que lhes cabem. Porém, a principal argumentação é o trabalho em conjunto com outros pesquisadores do grupo que desenvolveram atividades nessas linhas e partiram de princípios científicos práticos e inovadores no departamento.

Com isso, verificou-se que os resultados cumpriram os objetivos propostos, comprovando-se na prática através das simulações de problemas considerados de alta tecnologia no eletromagnetismo computacional e que são explorados internacionalmente, obtendo-se bom desempenho computacional para problemas de médio porte a um baixo custo financeiro. Outra importante contribuição foi o conhecimento construído em computação paralela que possibilitará o desenvolvimento de trabalhos futuros junto ao grupo, visando tornar as soluções numéricas mais ágeis.

5.2 Trabalhos Futuros

Apesar dos bons resultados de desempenho já mostrados, ainda existem possibilidades de melhorias no desenvolvimento de códigos como, por exemplo, a utilização da alocação dinâmica dos domínios computacionais e a investigação e implementação de técnicas de otimização dos algoritmos.

Continuar o desenvolvimento da interface com os usuários é um trabalho importante, pois permitirão que eles informem mais facilmente às estruturas necessárias as suas simulações numéricas. Esse tipo de ferramenta já existe comercialmente, porém os custos são muito elevados e o suporte, na grande maioria, ocorre em alguma língua estrangeira, motivando ainda mais a produção de um aplicativo doméstico.

A implementação de outros métodos numéricos em paralelo é interessante para a exploração de estruturas mais complexas que exijam soluções mais precisas que as vias FDTD como, por exemplo, malhadores para geometrias aleatórias onde, em alguns casos, a visualização dos resultados pode demorar dias para serem obtidos.

Referências Bibliográficas

- [1] J. Werthein, “*A sociedade da informação e seus desafios*”, Ciência da Informação, Brasília, v. 29, no. 2, pp. 71-77 (2000)
- [2] K. P. Fróiz, “*Uma breve história do fim das certezas ou paradoxo de Janus*”, Cadernos de pesquisa interdisciplinar em ciências humanas, no. 63 (2004)
- [3] K. Claudio, “*Resolução de sistemas de equações lineares provenientes da simulação de estruturas fotônicas*”, dissertação de mestrado, Departamento de Microonda e Óptica (DMO), Faculdade de Engenharia Elétrica e de Computação(FEEC), Universidade Estadual de Campinas (Unicamp), Agosto (2003)
- [4] <http://www.beowulf.org> (acessado em 19/05/2005)
- [5] <http://www.mct.gov.br/temas/si/default.asp> (acessado em: 10/05/2005)
- [6] M. Castells, “*A Sociedade em Rede*”, Editora Paz e Terra, São Paulo (1999)
- [7] S. Humphries, *Field Solutions on Computer*”, CRC Press LLC, Florida (2000)
- [8] J. J. Dongarra, *et al*, “*Numerical Linear Algebra for High-Performance Computers*”, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, (1998)
- [9] P.Veríssimo, L. Rodrigues, *Distributed Systems for System Architects*, Kluwer Academic Publisher, Massachusetts (2001)
- [10] <http://www.ic.unicamp.br/~ranido/mc964/pdf/Aula1.pdf> (acessado em: 19/05/2005)
- [11] *IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture*, The Institute of Electrical and Electronics Engineers, Incorporated – IEEE (2002)
- [12] A. S. Tanenbaum, M. van Steen, *Distributed Systems Principles and Paradigms*, Prentice Hall, New Jersey (2002)
- [13] W. C. L. Scofield, “*Uma Implementação Híbrida e Distribuída do Problema Job-Shop através dos algoritmos GRASP e Genético – Relatório Técnico*”, DECOM-UFOP (2002)
<http://www.decom.ufop.br/prof/marcone/Disciplinas/InteligenciaComputacionalRelatorioTecnicoWendrer.pdf> (acessado em 15/01/2005)

- [14] <http://www.lam-mpi.org/papers/delivery.paper/> (acessado em: 11/05/2005)
- [15] H. Zima, B. Chapman, *Supercompilers for Parallel and Vector Computers*, ACM Press, New York, (1991)
- [16] N. S. Simões, S. F. Souza, L. Muniz, F. Fardim Junior, A. F. Souza, Reis N. C. Junior, A. M. P. Valli, L. Catabriga, “*Instalação e Configuração de Clusters de Estações de Trabalho: Experiência do Laboratório de Computação de Alto Desempenho do Departamento de Informática da UFES*”, IV Workshop em Sistemas Computacionais de Alto Desempenho - WSCAD'2003, São Paulo (2003)
- [17] M. Pitanga. *Construindo Supercomputadores em Linux Segunda Edição*, Editora Brasport, São Paulo, (2004)
- [18] <http://www.beowulf.org/overview/history.html> (acessado em: 19/05/2005)
- [19] C. A. F. Oliveira, “*Cluster Beowulf- uma solução de baixo custo*”, Revista Científico, Ano IV, V. I, Salvador, (2004)
- [20] <http://www.beowulf.org/overview/faq.html#17> (acessado em: 19/05/2005)
- [21] M. Pourzandi, et al., “*A Distributed Security Infrastructure for Carrier Class Linux Clusters*”, Proceedings of 2002 Ottawa Linux Symposium, Ottawa (2002)
- [22] N. Gorsuch, N. “*Linux Cluster Security*”, 2º LCI International Conference on Linux Clusters, University of Illinois (2001)
- [23] N. Gorsuch, “*The PFILTER Firewall Compiler for Cluster*”, 1º OSCAR Symposium, Quebec (2003)
- [24] W. Yucik, G. A. Koenig, X. Meng, J. Greemseid, “*Cluster Security as a Unique Problem with Emergent Properties: Issues and Techniques*”, 5º LCI International Conference on Linux Clusters, Austin, Texas (2004)
- [25] M. N. O.Sadiku, *Elements of Electromagnetics, Third Edition*, Oxford University Press, New York (2001)
- [26] <http://rsrawat.myplace.nie.edu.sg/Teaching/CAP102/CAP%20102-Lecture7-Interference.htm> (acessado em: 19/05/2005)
- [27] Y. Geng, X. Wu P. Jiao, “*FDTD Analysis of the interaction between Electromagnetic wave and inhomogenous anisotropic Media*”, Antennas, Propagation and EM Theory, 2000. Proceedings. ISAPE 2000. 5th International Symposium, pp 491 - 495 (2000)

- [28] K. S. Yee, "Numerical Solution of Initial Boundary Value Problems Involving Maxwell's Equations in Isotropic Media" – IEEE Transactions Antennas and Propagation, Vol. 14, pp: 302-307 (1966)
- [29] L. Sandrini, A. Vaccari, C. Malacarne, L. Cristoforetti, R. Pontalti, "RF dosimetry: a comparasion between power absorption of female and male numerical models from 0.1 to 4 GHz", Physics in Medicine and Biology, Vol. 49, pp: 5185-5201 (2004)
- [30] W. Kainz, D. D. Chan, J. P. Casamento, H. I. Bassen, "Calculation of induced current densities and specific absorption rates (SAR) for pregnant women exposed to hand-held metal detectors", Physics in Medicine and Biology, Vol. 48, pp 2551-2560 (2003)
- [31] L. A. Ambrosio, "Cálculo da Taxa de Absorção Específica (SAR) na Cabeça Humana Através do Modelo de Campos-de-Onda Quiral", dissertação mestrado, Departamento de Microonda e Óptica (DMO), Faculdade de Engenharia Elétrica e de Computação(FEEC) da Universidade Estadual de Campinas (Unicamp), Campinas (2005)
- [32] R. O. Santos, C. L. S. Sobrinho, "FDTD method: Analysis of one-dimensional array of H-plane sectorial horn antennas with dielectric lens" Revista Científica Periódica - Telecomunicações Vol. 06, no. 01 (2003)
- [33] J. S. Araújo, et al. "Analysis of Antennas by FDTD Method Using Parallel Processing with MPI", International Microwave and Optoelectronics Conference, IMOC, Foz do Iguaçu (2003)
- [34] U. Oguz, L. Gürel, "Modeling of Ground-Penetrating-Radar Antennas With Shields and Simulated Absorbers", IEEE Transactions on Antennas and Propagation, Vol. 49, no. 11, pp: 1560-1567 (2001)
- [35] S. Xu, P. Zhou, "FDTD Analysis for Satellite BFN Consisting of Rectangular Coaxial Lines", Asia Pacific Microwave Conference (1997)
- [36] M. F. Su, I. El-Kady, D. A. Bader, S. Lin, "A Novel FDTD Application Featuring OpenMP-MPI Hybrid Parallelization", Proceedings of the 2004 International Conference on Parallel Processing, Computer Society, pp: 373-379 (2004)
- [37] W. C. Chew, J. M. Jin, C. C. Lu, E. Michielssen, J. M. Song "Fast Solution Methods in Electromagnetics", IEEE Transactions Antennas and Propagation, vol. 45, no. 3, pp: 533-543(1997)
- [38] A. Taflove, S. C. Hagness, *Computational Electrodynamics: The Finite-Difference Time-Domain Method, 2nd. Edition*, Artech House, Norwood (2000)
- [39] A. F. Peterson, S. L. Ray, R. Mittra, *Computational Methods for Electromagnetics, Series on Electromagnetic Waves*, IEEE Press, New York (1998)

- [40] J. S. Araújo, *et al.*, “Análise de Antenas em 2-D Utilizando o Método das Diferenças Finitas no Domínio do Tempo com Processamento Paralelo e MPI”, Revista Científica Periódica - Telecomunicações, Vol. 06, no. 06, pp: 18-28 (2003)
- [41] T. G. Moore, J. G. Blaschak, A. Taflove, G. A. Kiegsman, “*Theory and Applications of Radiation Boudary Operators*”, IEEE Transactions on Antennas and Propagation, Vol. 36, no. 12, December (1988)
- [42] J. P. Berenger, “*A Perfect Matched Layer for the Absorption of Electromagnetic Waves*”, Journal of Computational Physics, 114, pp. 185-200 (1994)
- [43] E. J. Sartori, “*Estudo Experimental de Metamateriais Baseados em Grades Dielétricas*”, dissertação de mestrado, Departamento Microondas e Óptica (DMO), Faculdade de Engenharia Elétrica e de Computação (FEEC) da Universidade Estadual de Campinas (Unicamp), Campinas (2004)
- [44] D. S. Katz, E. T. Thiele, A. Taflove, “*Validation and Extension to Three Dimensions of the Berenger PML Absorbing Boundary Condition for FD-TD Meshes*”, IEEE Microwave and Guided Wave Letters, Vol. 4, no. 8, pp: 268-270 (1994)
- [45] J. P. Bérenger, “*Perfectly Matched Layer for the FDTD Solution of Wave-Structure Interaction Problems*”, IEEE Transactions on Antennas and Propagation, Vol. 44, no. 1, pp: 110-117 (1996)
- [46] A. Gueist, *et al.*, “*PVM 3 User’s Guide and Reference Manual*”, OAK Ridge Laboratory, (1994)
- [47] J. J. Dongarra, S. W Otto, M. Snir, D. Walker, “*An Introduction to MPI Standard*”, <http://www.netlib.org/utk/people/JackDongarra/pdf/ut-cs-95-274.pdf>, (acessado em: 20/08/2005) (1995)
- [48] <http://www.gnu.org/copyleft/gpl.html> (acessado em: 18/05/2005)
- [49] J. M. Laine, “*Desenvolvimento de Modelos para Predição de Desempenho de Programas Paralelos MPI*”, Dissertação de Mestrado, Escola Politécnica, Universidade de São Paulo (2003).
- [50] <http://gcc.gnu.org/> (acessado em: 19/05/2005)
- [51] <http://www.lam-mpi.org/> (acessado em: 19/05/2005)
- [52] “*MPI: A Message Passing Interface Standard – Message Passing Interface Forum*”, supported by ARPA and NSF (1995)
- [53] <http://www.lam-mpi.org/about/devel/> (acessado em: 19/05/2005)

- [54] B.H. Bennett, E. Davis, T. Kunau, "*Beowulf Parallel Processing for Dynamic Load-balancing*", Proceedings of the IEEE Aerospace Conference, pp: 389-395 (2000)
- [55] A. Taflove, "*Why Study Electromagnetics: The First Unit in an Undergraduate Electromagnetics Course*", IEEE Antennas and Propagation Magazine, vol. 44, no. 2, pp:132-139 (2002)
- [56] M. L. Zamorano, H. S. Torres, M. E. Soto, "*Modelado y Simulación del SAR Debido a los Campos Electromagnéticos Radiados por Teléfonos Celulares*", XI Congreso Internacional de Telecomunicaciones y Muestra Tecnologías UACH, Valdivia-Chile (2004)
- [57] A. A. A. Sales, C. R. Fernández, M. Bonadiman, "*Simulações da SAR na cabeça e antenas planares para telefones móveis*", *Revista Brasileira de Engenharia Biomédica*, vol. 19, no. 2, pp. 77-90, (2003)
- [58] V. G. Veselago, "*The Electrodynamics of Substance With Simultaneously Negative Values of ϵ e μ* ", Soviet Physics Uspekhi, Vol. 10, no. 4, pp 509-514 (1968)
- [59] I. V. Shadrivov, *et al.*, "*Transmission properties of left-handed band-gap structures*", arXiv:physics/0402087 v1, (2004)
- [60] J. B. Pendry, J. Holden, D. J. Robbins, W. J. Stewart, "*Magnetism from Conductors and Enhanced Nonlinear Phenomena*", IEEE Transactions on Microwave Theory and Techniques, Vol. 47, no. 11, pp:2075-2084 (1999)
- [61] R. W. Ziolkowski, E. Heyman, "*Wave propagation in media having negative permittivity and permeability*", Physical Review E, Vo. 64, pp: 056625-1 – 056625-15(2001)
- [62] A. M. P. A. Silva, "*Tecnologia de Estruturas EBG-Electromagnetic Bandgap – Aplicada a Antenas para Comunicações Móveis*", Departamento de Microondas e Óptica (DMO), Faculdade de Engenharia Elétrica e de Computação (FEEC), Universidade Estadual de Campinas (2004)

Anexos

Anexo A

Instalando e Configurando um Cluster Beowulf

A instalação de um *cluster Beowulf* não é complexa de ser realizada, porém demanda um bom trabalho, aumentando de acordo com a quantidade de equipamentos, caso não haja um procedimento de automatização para a implantação. Na Internet, é possível encontrar muito material a respeito; entretanto, de acordo com a busca realizada, não foi encontrado nenhum tutorial que utilizasse a autenticação através do SSH2 e com a chave do tipo DSA, que acrescentam importantes fatores de segurança ao sistema.

Nesta seção será apresentado como implementar o *cluster*, utilizando mecanismos de segurança de rede.

Configurando o *Cluster Beowulf*

Levando em consideração que já se conhece a estrutura e disposição dos computadores, o passo seguinte é detalhar quais os níveis de acesso para a delimitação dos itens de segurança a serem utilizados. Nesse projeto, a rede está numa faixa de endereços inválidos na Internet, ou seja, as máquinas dessa *intranet* não são acessíveis pela Internet, sendo necessário passar um *gateway* (máquina que interliga duas redes diferentes); a disposição dos *clusters* (Fig. A1) implementados seguem a mesma idéia, independente das configurações desses dispositivos, como processadores e memórias.

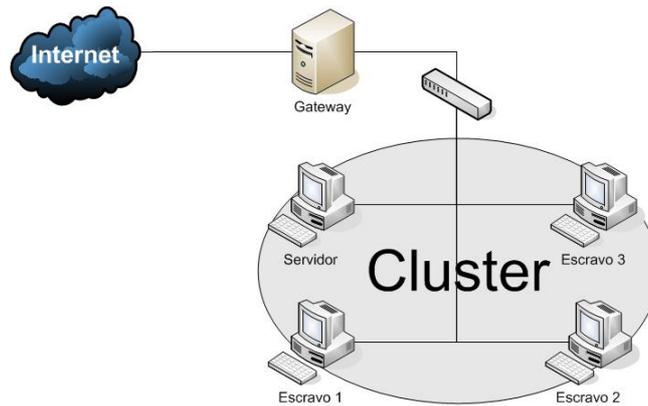


Fig. A1. – Representação de um cluster

As configurações são diferenciadas entre o nó mestre (servidor) e os escravos. Portanto, os detalhes a seguir sobre as configurações do *cluster* estão devidamente separados e identificados, porém em todas as máquinas deverão existir usuários com o mesmo nome como, por exemplo, *matrix*. Esse usuário tem seu diretório padrão em */home/matrix*. Além disso, serão necessários pacotes para o uso do NFS (*Network File System*) e da biblioteca para troca de mensagens que, no nosso caso, optou-se pela LAM-MPI (*Local Área Multicomputer-MPI*).

Para ativar o NFS em todas as máquinas, informe o comando `ntsysv`, e selecione o NFS no menu que aparece.

Nó Mestre (servidor)

1. Inserir os endereços e nomes das máquinas no */etc/hosts*; caso não saiba os nomes das máquinas, informe o comando `hostname` no *Shell* do sistema operacional e a seguir preencha o arquivo como segue no exemplo:

10.0.06	Servidor
10.0.07	Esclavo1
10.0.08	Esclavo2
10.0.09	Esclavo3
10.0.10	Esclavo4

2. Como super-usuário (root), configurar NFS através do arquivo */etc/matrix*; veja abaixo o modelo:

```
/home/matrix/MPI 10.0.0.101(rw)
/home/matrix/MPI 10.0.0.102(rw)
/home/matrix/MPI 10.0.0.103(rw)
/home/matrix/MPI 10.0.0.104(rw)
/home/matrix/MPI 10.0.0.105(rw)
/home/matrix/MPI 10.0.0.106(rw)
/home/matrix/MPI 10.0.0.107(rw)
/home/matrix/MPI 10.0.0.108(rw)
/home/matrix/MPI 10.0.0.109(rw)
/home/matrix/MPI 10.0.0.110(rw)
```

3. Mude o usuário em uso, seguindo o caso para esse exemplo, informando o comando `su matrix`.
4. Crie a chave de assinatura digital através do comando: `ssh-keygen -t dsa`, note que no diretório *.ssh* do diretório padrão do usuário foi criado o arquivo *id_dsa.pub*, que é chave pública para o acesso remoto sem senha, porém com a segurança de ser uma chave criptografada.
5. Envie esse arquivo a todas as máquinas do *cluster*, acessando-as com o usuário de mesmo nome como, por exemplo, `ssh matrix@escrav01`.

Nós Escravos

1. Como super-usuário nos nós escravos, primeiramente é necessário configurar o arquivo */etc/fstab*, para a montagem automática das partições compartilhadas no servidor, ou seja, a partição do disco de armazenamento do servidor estará acessível nos nós escravos. Para configurar, siga o modelo apresentado abaixo:

2. Trocar de usuário, indo para o padrão do *cluster*, que no caso desse exemplo, foi adotado o *matrix*.

3. No diretório padrão do usuário, verificar se o diretório *.ssh* já existe, caso não, execute o seguinte comando:

```
ssh henrique@servidor
```

4. A seguir, execute o comando abaixo para validar a chave pública do servidor nos nós escravos, permitindo o acesso remoto do servidor aos outros nós sem senha:

```
cat id_dsa.pub >> .ssh/authorized_keys
```

5. Altere as permissões do diretório e do arquivo, como segue:

```
chmod 700 .ssh
```

```
chmod 644 .ssh/authorized_keys
```

6. Para testar esse procedimento, tente acessar do servidor qualquer outro nó escravo já configurado através do comando:

```
ssh henrique@escrav01
```

Com isso, os dois tipos de nós de um *cluster Beowulf* estão configurados e prontos para serem utilizados para troca de mensagens. Uma recomendação aos iniciantes é reiniciar os equipamentos, primeiramente o servidor e esperar que ele esteja em funcionamento. A seguir, religar os nós escravos para verificar se o acesso via *ssh* sem senha está funcionando e se realmente a partição está montada corretamente no diretório desejado.

Para acionar os processos em todas as máquinas é preciso criar um arquivo (*lamhosts*) com o nome das máquinas e, a seguir, informar o seguinte comando:

```
lamboot -v lamhosts
```

O termo *-v* é importante para que seja mostrado na tela qualquer problema que possa surgir durante o início dos processos.

Anexo B

Instalando, Configurando e Utilizando a LAM-MPI

A biblioteca LAM-MPI (*Local Area Multicomputer – Message Passing Interface*) é desenvolvida através do padrão MPI-2 e, atualmente, está sob a responsabilidade da *Open Systems Laboratory* (OSL), da Universidade de Indiana-E.U.A.. Essa é uma biblioteca para computação de alto desempenho, de uso gratuito e de código aberto para implementações. Ela é passível de ser utilizada em simples *cluster* até máquinas com multiprocessamento simétrico (SMP) em redes velozes, e mesmo em ambientes heterogêneos.

Esta biblioteca apresenta inúmeros recursos que são mencionados no sítio do projeto (<http://www.lam-mpi.org/about/overview/>), sendo importante ressaltar que ela cobre todas as especificidades do MPI-1 e a maioria dos recursos padronizados no MPI-2, sendo importante destacar a possibilidade de trabalho com *threads*, interoperabilidade entre linguagens de programação (C, C++ e Fortran), pontos de checagens dos processos, a comunicação entre os processos apresenta alto desempenho devido ao baixo *overhead* na pilha TCP e suporte a SMP. Todos esses recursos contribuem bastante em trabalhos de paralelização, devido à qualidade e quantidade de recursos disponíveis.

Ainda não foram mencionadas as funções de troca de mensagens propriamente ditas. No endereço <http://www.lam-mpi.org/tutorials/bindings/>, da Internet, é possível encontrar uma lista completa das funções contidas na LAM-MPI. Dentre essas é importante comentar algumas delas que são básicas em grande parte dos aplicativos paralelos, sendo:

MPI_COMM_RANK(comm, rank)

Atribui a *rank*, um valor identificando no número do processo

MPI_COMM_SIZE(comm, size)

Atribui a *size*, a quantidade de processos em execução

MPI_FINALIZE()

Finalizar os processos MPI

MPI_INIT()

Iniciar o MPI

MPI_RECV(buf, count, datatype, source, tag, comm, status)

Receber mensagens de outros processos

MPI_SEND(buf, count, datatype, dest, tag, comm)

Enviar mensagens a outros processos

Essas funções foram utilizadas nos aplicativos desenvolvidos. Para exemplificar o uso abaixo segue um aplicativo bem simples, em que o processo inicial (0) envia aos outros processos a mensagem “Olá Mundo”, que recebem esta mensagem e imprimem na tela.

```
#include <stdio.h>
#include <stdlib.h>
#include "mpi.h"
int main(int argc, char ** argv)
{
    int id_processo, qtd_processo, i;
    char * texto;
    MPI_Status status;
    MPI_Init (&argc, &argv);
    MPI_Comm_rank (MPI_COMM_WORLD, &id_processo);
    MPI_Comm_size (MPI_COMM_WORLD, &qtd_processo);
    texto = (char*) malloc (sizeof(char) * 30);

    if (id_processo==0)
    {
        strcpy(texto, "Olá Mundo");
        for (i=1; i < qtd_processo; i++)
            MPI_Send(texto, 30, MPI_CHAR, i, 1,
MPI_COMM_WORLD);
    }
    else
    {
        MPI_Recv(texto, 30, MPI_CHAR, 0, 1, MPI_COMM_WORLD,
&status);
        printf ("%s Processo %i \n", texto, id_processo);
    }
    MPI_Finalize();
    return 0;
}
```

Para a execução desse programa é necessário que a biblioteca LAM-MPI esteja devidamente instalada nos equipamentos, assim como suas dependências.

A verificação, no sistema operacional Linux, pode ocorrer através do comando:

```
rpm -q lam
```

Caso não esteja instalada, é possível obtê-la através do seguinte endereço <http://www.lam-mpi.org/7.1/download.php> na Internet, ressaltando a importância de se atentar a versão de interesse. Uma dependência para a execução de aplicativos MPI, através da biblioteca LAM-MPI é o uso da *libaio-devel* que está disponível no disco 4 do Fedora (Core 2 ou 3). Para a instalação dos pacotes utilize o seguinte comando:

```
rpm -ivh <nome_pacote>
```

Compilando e executando aplicativos

Para verificar se a LAM-MPI está funcionando no computador copie e compile o código acima apresentado, através do comando:

```
mpicc teste.c -o teste
```

Inicie a biblioteca para processar as comunicações de processos; para isso crie um arquivo com os nomes das máquinas (*hostname*); em nosso exemplo, utilizamos o arquivo lamhosts

```
lamboot -v lamhosts
```

Verifique se o arquivo binário foi gerado corretamente e a seguir acione-o por:

```
mpirun C teste
```

Esse comando executa os programas MPI, possibilitando vários recursos como, por exemplo, em uma única máquina, com um único processador, é possível simular vários processos, sendo:

```
mpirun -np <numero_de_processos> teste
```

Os recursos disponíveis, a cada comando, podem ser acessados no próprio manual contido no sistema operacional, através do comando:

```
man <nome_do_comando>
```

Anexo C

Leis de Amdahl

A chamada “Lei de Amdahl” foi idealizada pelo arquiteto de computação Gene Myron Amdahl, que era um dos principais conhecedores da linha de supercomputadores da IBM (International Business Machine), sendo esta fundamental para a teoria da computação paralela. Essa lei demonstra os ganhos de desempenho obtidos com o processamento paralelo, utilizando dados de aplicativos seqüenciais em comparação com os paralelos, analisando o *speedup* obtido.

O *speedup* mede o quanto mais rápido o código paralelo é mais rápido que o seqüencial, sendo definido pela seguinte fórmula:

$$S_p = \frac{T_1}{T_p} \quad (1)$$

onde:

- p é número de processos
- T_1 é o tempo de execução do algoritmo seqüencial
- T_p é o tempo de execução do algoritmo paralelo com p processos

O *speedup* ideal, ou linear, é obtido pela relação $S_p = p$; isso ocorre, por exemplo, quando a tarefa realizada em dois processos é duas vezes mais rápida que quando executada em um único processo, sendo isto considerado uma excelente escalabilidade. A escalabilidade em telecomunicações e engenharia de *software*, indica a capacidade de um sistema melhorar o

desempenho quando os recursos (geralmente *hardware*) são melhorados e/ou adicionados.

Para a medição da eficiência que os sistemas paralelos proporcionam, é necessário dividir o *speedup* pela quantidade de processos, ou seja,

$$E = \frac{S_p}{p} \quad (2)$$

Sendo este um valor tipicamente entre zero e um. A finalidade desse cálculo é estimar o quão bem estão sendo utilizados os processos para solucionar o problema, comparando com a quantidade de esforços necessários para a comunicação e sincronização.

Assim, pode-se escrever a Lei de Amdahl como sendo:

$$Am = \frac{1}{(1 - S_p) + \frac{S_p}{p}} \quad (3)$$

Com isso, tem-se os cálculos necessários à obtenção das relações de desempenho do processamento paralelo, perfeitamente aplicáveis nas análises deste trabalho, o qual é executado numa arquitetura de *cluster Beowulf* heterogêneo.