

**Pontifícia Universidade Católica do Rio Grande do Sul**

**Faculdade de Engenharia**

**Programa de Pós - Graduação em Engenharia Elétrica**

**INTERAÇÃO ENTRE ONDAS ELETROMAGNÉTICAS E  
CORPOS IRRADIANTES E IRRADIADOS**  
**Simulação pelo Método FDTD**

Orientado

Eng. Leandro Bauermann Porfírio

Orientador

Prof. Dr. Eng Rubem Dutra Ribeiro Fagundes

Porto Alegre, Março de 2007

## **Dedicatória**

Este trabalho é dedicado a ...

... minha esposa Adriane, por me apoiar, me incentivar e suportar os inúmeros dias e noites em que apenas meu corpo estava presente

... minha mãe Marlena, por me emprestar um pouco de sua sabedoria sempre que precisei e por me apoiar incondicionalmente na realização de meus projetos de vida

... ao meu filho Roberto, que está por vir, o maior presente que eu poderia receber

## **Agradecimentos**

Desejo registrar o agradecimento ao Prof. Dr. Eng. Fernando César Comparsi de Castro, à Prof. Dra. Eng. Maria Cristina Felippetto de Castro e ao Prof. Dr. Eng. Rubem Dutra Ribeiro Fagundes por suas contribuições no trabalho, pela dedicação e atenção na orientação e pelo apoio necessário para o desenvolvimento do tema.

# Sumário

<b>1. Resumo</b>	<b>5</b>
<b>2. Abstract</b>	<b>6</b>
<b>3. Objetivos</b>	<b>7</b>
<b>3.1 Justificativa</b>	<b>7</b>
<b>3.2 Apresentação</b>	<b>9</b>
3.2.1 Noções sobre o Método dos Momentos	9
3.2.2 Introdução ao FDTD	10
<b>4. Fundamentação Teórica</b>	<b>15</b>
<b>4.1 O Algoritmo de Yee</b>	<b>16</b>
<b>4.2 Desenvolvimento matemático</b>	<b>18</b>
<b>4.3 Tamanho de célula e incremento de tempo</b>	<b>22</b>
<b>4.4 Fontes de excitação</b>	<b>22</b>
<b>4.5 Equacionamento para cálculo numérico</b>	<b>24</b>
<b>4.6 ABC (Absorbing Boundary Conditions)</b>	<b>25</b>
4.6.1 Métodos Analíticos	27
4.6.2 Métodos Numéricos	28
4.6.3 Meios Externos Absorventes (PML e UPML)	31
<b>5. Proposta de Trabalho</b>	<b>44</b>
<b>6. Metodologia</b>	<b>45</b>
<b>6.1 Hipóteses ao equacionamento</b>	<b>45</b>
<b>6.2 Algoritmo básico e recursos computacionais</b>	<b>45</b>
6.2.1 Organização de variáveis por plano espacial	49
<b>6.3 Algoritmo completo do FDTD</b>	<b>57</b>
6.3.1 Implementação do algoritmo de Yee sem UPML	57
6.3.2 Implementação da UPML	59
<b>7. Resultados</b>	<b>65</b>
<b>7.1 Tempo de processamento</b>	<b>66</b>
<b>7.2 Otimização do acesso ao disco</b>	<b>67</b>
<b>7.3 Ensaios de propagação e reflexão da onda nas bordas</b>	<b>68</b>
<b>8. Conclusão</b>	<b>94</b>
<b>Referências Bibliográficas e Outras</b>	<b>96</b>

## 1. Resumo

O propósito da dissertação é abordar a simulação numérica, no domínio tempo, da propagação no espaço tridimensional de ondas eletromagnéticas e sua interação com objetos.

Mais precisamente, foi realizada a implementação do algoritmo de Yee, incluindo técnicas para minimizar os problemas de reflexão numérica que ocorrem nas bordas do espaço tridimensional de simulação.

O resultado do trabalho é um software genérico que realiza a simulação em quatro dimensões (três espaciais e uma temporal), executável em computadores de uso geral e preparado para tratar problemas complexos (de geometria complexa, com diversos materiais diferentes).

Em outras palavras, tem-se a implementação numérica de uma câmara anecóica, que permite simular as interações de ondas eletromagnéticas e os objetos no interior interior da câmara anecóica.

Foi apresentada uma nova técnica de programação, denominada “organização de variáveis por plano espacial”, que permitiu a realização do trabalho proposto.

## **2. Abstract**

*Time domain numerical simulation of the electromagnetic waves propagation in a tridimensional space and its interaction with objects is the crue of the contributions of this work*

*More specifically, the implementation of the Yee's algorithm was carried out, including techniques to minimize the problems of numerical reflection that occur in the edges at the tridimensional space of simulation.*

*The result of this work is a generic software that simulates in four dimensions (three in space and one in time). This software can handle complex problems (complex geometry and diverse different materials) and is executable in a general purpose computers.*

*In other words, a numerical implementation of an anechoic chamber allows the simulation of the interactions of electromagnetic waves and the objects inside of anechoic chamber.*

*A new programming technique called "variables organization in space plans", t allowed the accomplishment of this work.*

### 3. Objetivos

A meta da Dissertação é o desenvolvimento de uma ferramenta computacional para a simulação de elementos irradiantes e da interação de ondas eletromagnéticas em um meio composto de diversos materiais, baseada no método das **Diferenças Finitas no Domínio do Tempo** (*The Finite-Difference Time-Domain Method*) ou **FDTD**.

Mais especificamente, implementar e testar o algoritmo de Yee e as técnicas necessárias para eliminar os problemas de reflexão numérica que ocorrem nas bordas do espaço de simulação, denominadas em uma tradução livre de Condições de Contorno Absorventes (*ABC – Absorbing Boundary Conditions*).

Além de testar o algoritmo de Yee num espaço tridimensional e a principal técnica de ABC atualmente adotada, pretende-se propor uma nova técnica de ABC e comparar sua performance com a dos demais algoritmos implementados.

Embora não pareça a princípio uma tarefa árdua, ao longo dessa apresentação será possível vislumbrar as dificuldades que são esperadas e a complexidade do tema.

Por último, apesar de não ter sido objeto direto da Dissertação, o trabalho realizado envolveu o desenvolvimento de uma nova técnica de programação, a qual poderá ter utilização em algoritmos outros além do FDTD.

#### 3.1 Justificativa

A evolução dos sistemas digitais aplicados ao processamento de dados e às telecomunicações tem exigido um conhecimento cada vez maior da interação entre os elementos irradiantes presentes.

Os equipamentos microprocessados, entre os quais se destacam os computadores de uso geral (PCs), têm utilizado sinais de frequência cada vez maior na realização de suas tarefas. É comum serem encontrados no mercado PCs

trabalhando com sinais de 1 GHz. Nessa faixa de frequência e acima, qualquer trilha da placa de circuito impresso é um elemento irradiante, cuja influência nas trilhas e em componentes vizinhos pode ser significativa e deve ser avaliada, a fim de prever e evitar os erros associados a esse fenômeno.

Na telefonia celular, o desejo de otimizar os sistemas atuais e de viabilizar serviços de comunicação de dados de alta performance, requerem elementos irradiantes (antenas) cada vez mais sofisticados. Antenas que permitam a alteração dinâmica de seu diagrama de irradiação, de forma a concentrar o campo na posição de cada usuário móvel (antenas adaptativas), são vistas como a solução para o problema.

Ainda, a transmissão de informações empregando técnicas de comunicação como a *Ultra Wide Band* (UWB) levam os sistemas aos limites de velocidade, trabalhando com pulso de duração extremamente pequena. Nesse caso, as hipóteses necessárias para avaliar o sistema no domínio da frequência perdem significado, devido à necessidade de avaliar o comportamento transitório do sistema e não apenas seu estado em regime.

Além disso, verifica-se uma crescente utilização pela medicina de procedimentos não invasivos para o diagnóstico e tratamento de doenças. A maioria dessas técnicas envolve a interação de ondas eletromagnéticas e os diversos tecidos que compõe o organismo. A otimização das técnicas atuais e o desenvolvimento de novas técnicas requerem meios para simular tal interação.

Esse cenário justifica o interesse em desenvolver ferramentas que permitam a simulação de elementos irradiantes e da interação de ondas eletromagnéticas com os diversos materiais que compõe um meio.

Nesse campo da engenharia, algumas teorias foram desenvolvidas para embasar a métodos computacionais que realizem tal implementação. Entre esses, devido a sua generalidade na abordagem e flexibilidade em sua utilização, destaca-se o método das **Diferenças Finitas no Domínio do Tempo** (*The Finite-Difference Time-Domain Method*) ou **FDTD**.

## 3.2 Apresentação

Ao longo do tempo, os problemas de modelamento de elementos irradiantes e da interação entre ondas eletromagnéticas e corpos não transparentes tiveram duas abordagens matemáticas distintas para sua solução numérica, conforme as equações que modelam o fenômeno:

- a) solução de equações integrais, normalmente tratadas no domínio da frequência. Nesse grupo, destaca-se o Método dos Momentos (MoM).
- b) solução de equações de diferenças parciais, nesse acaso no domínio do tempo, onde se enquadra o FDTD.

### 3.2.1 Noções sobre o Método dos Momentos

O método MoM foi desenvolvido por B.G. Galerkin, em 1915 na Rússia, e por Krylov e Kravchuk, entre 1926 e 1932 na ex-União Soviética). Porém, tornou-se popular após a publicação do livro de Harrington em 1968 [3].

Como método MoM não será abordado adiante, segue uma breve explicação da teoria envolvida, a fim de ter-se uma noção de como o modelamento é realizado.

O objetivo do método é avaliar a impedância e campo elétrico associados a um objeto problema qualquer, quando excitado ou irradiado. Para tanto, busca-se resolver o problema a partir da seguinte equação, avaliada ao longo de toda superfície problema:

$$\mathbf{E}^{total}(\mathbf{r}) = \mathbf{E}^{inc}(\mathbf{r}) + \mathbf{E}^s(\mathbf{r}) \quad (\text{Eq.1})$$

onde:

$\mathbf{r}$  – vetor posição

$\mathbf{E}^{total}$  – fasor campo elétrico total na superfície

$\mathbf{E}^{inc}$  – fasor campo elétrico incidente na superfície (conhecido)

$\mathbf{E}^s$  – fasor campo elétrico espalhado pela superfície

Sabe-se que o campo total  $\mathbf{E}^{total}(\mathbf{r})$  está associado a densidade de corrente de polarização  $\mathbf{J}_{pol}(\mathbf{r})$ , por uma expressão que envolve apenas as propriedades elétricas do material e a frequência de propagação da onda, que está abaixo apresentada:

$$\mathbf{J}_{pol}(\mathbf{r}) = [\sigma + j\omega\epsilon_0(\epsilon_r - 1)]\mathbf{E}^{total}(\mathbf{r}) \quad (\text{Eq.2})$$

onde:

$\sigma$  – condutividade elétrica

$\omega$  – frequência angular de propagação da onda

$\epsilon_r$  – permissividade elétrica do material, relativa ao vácuo

$\epsilon_0$  – permissividade elétrica no vácuo

Ainda, conforme modelo, o campo espalhado  $\mathbf{E}^s$  é avaliado em um ponto de observação arbitrário, fora da superfície problema, e pode ter duas origens distintas, conforme a natureza da superfície problema:

- a) condutor perfeito – o fasor  $\mathbf{E}^s$  é avaliado por uma integral de superfície, que envolve a função de Green e a densidade de corrente na superfície  $\mathbf{J}_s(\mathbf{r})$ . Nesse caso, tem-se  $\mathbf{E}^{total} = \mathbf{0}$ , pois não existe corrente de polarização (a corrente está na superfície do condutor).
- b) material dielétrico – o fasor  $\mathbf{E}^s$  é avaliado por uma integral de volume, também envolvendo a função de Green e densidade de corrente na polarização  $\mathbf{J}_{pol}(\mathbf{r})$

Assim, para resolver a equação integral, o método prevê o desdobraamento do material em elementos, adotando uma das seguintes alternativas:

- segmentos – para estruturas unidimensionais, como dipolos, e arranjos desses;
- malhas de segmentos – para superfícies condutoras;
- blocos – para materiais dielétricos tridimensionais.

Para completar o equacionamento, é estabelecida uma relação entre a impedância dos elementos (própria e mútua), o campo elétrico e a densidade de corrente, a qual nos conduz a um sistema de equações integrais a serem resolvidas simultaneamente.

### **3.2.2 Introdução ao FDTD**

O método FDTD foi apresentado por K. S. Yee em 1966 [1][2] e baseia-se na solução das equações de Maxwell em sua forma diferencial:

$$\frac{\partial \mathbf{H}}{\partial t} = \frac{1}{\mu} (-\nabla \times \mathbf{E} - \sigma^* \mathbf{H} - \mathbf{M}_{source}) \quad (\text{Eq.3})$$

$$\frac{\partial \mathbf{E}}{\partial t} = \frac{1}{\varepsilon} (\nabla \times \mathbf{H} - \sigma \mathbf{E} - \mathbf{J}_{source}) \quad (\text{Eq.4})$$

onde:

$\mathbf{H}$  – vetor campo magnético (função da posição e tempo)

$\mathbf{E}$  – vetor campo elétrico (função da posição e tempo)

$\varepsilon$  – permissividade elétrica do meio

$\mu$  – permeabilidade magnética do meio

$\sigma$  – condutividade elétrica do meio (coeficiente de atenuação do campo elétrico por atravessar o meio)

$\sigma^*$  – perda magnética equivalente (coeficiente de atenuação do campo magnético por atravessar o meio)

$\mathbf{J}_{source}$  – fonte de densidade de corrente elétrica

$\mathbf{M}_{source}$  – fonte de densidade de corrente magnética equivalente

Os vetores  $\mathbf{J}_{source}$  e  $\mathbf{M}_{source}$  atuam como fontes independentes de campo elétrico e magnético, as quais são inseridas no algoritmo de processamento após modelamento numérico [2]. Dessa forma, os vetores  $\mathbf{J}_{source}$  e  $\mathbf{M}_{source}$  serão inicialmente considerados nulos.

A solução simultânea dessas duas equações define o comportamento de uma onda eletromagnética qualquer no espaço e no tempo.

Dessa forma, a partir dessas equações, o método FDTD procura avaliar a evolução do estado do sistema (campos elétricos e magnéticos) em todos os pontos, ao longo tempo.

Assim, inicia-se o processo pela delimitação de um volume de controle, grande o suficiente para conter todos os corpos excitados e irradiados de interesse,

no interior do qual vai ser avaliado o comportamento dos campos elétricos e magnéticos ao longo do tempo.

Após, o volume de controle é dividido em células, pequenas o suficiente para assumir que as variações de campo de uma célula para outra são suficientemente pequenas, permitindo sua correta avaliação através das equações de Maxwell na forma diferencial. Para tanto, as dimensões da célula devem ser consideradas infinitesimais se comparadas com o comprimento de onda  $\lambda$  da eletromagnética em propagação.

A última etapa de preparação do volume de controle é a associação de cada célula as propriedades elétricas e magnéticas do material que a compõe ( $\epsilon$ ,  $\mu$ ,  $\sigma$  e  $\sigma^*$ ). A partir de então nosso volume de controle recebe a denominação de *grid*.

Antes de iniciar a avaliação da evolução do sistema ao longo tempo, deve-se estabelecer o intervalo de tempo  $\Delta t$  decorrido entre cada estado do sistema. Esse valor deve ser pequeno o suficiente para não permitir que a onda, em qualquer ponto do sistema e em qualquer direção, possa se propagar mais de uma célula por vez. Como a velocidade limite de propagação da célula é a velocidade da luz no vácuo, perceber-se que o valor de  $\Delta t$  é definido pelo tamanho da célula. Ou, em última análise, também depende de  $\lambda$ .

Agora, o processo numérico de avaliação da evolução do sistema pode ser iniciado. A cada intervalo  $\Delta t$ , todo o *grid* é varrido, ou seja, as equações diferenciais são aplicadas a cada célula do *grid*. Concluída a varredura, o tempo é acrescido de mais um período  $\Delta t$ , e inicia-se nova varredura.

Deve-se ressaltar, como pode-se perceber através da inspeção das equações de Maxwell, que cada célula interage apenas com as células vizinhas, o que facilita a solução numérica do problema.

As varreduras são encerradas quando for alcançado um determinado tempo final pré-estabelecido. Isso pode ser traduzido pela realização de um número "N" de varreduras, após as quais poderão ser apresentados os resultados de campo.

Outras características podem ser facilmente determinadas. Por exemplo: a distribuição de corrente é obtida da distribuição do campo magnético e a impedância é determinada pela razão entre campo elétrico e campo magnético.

Porém, o que distingue o FDTD dos demais métodos? Como resposta a tal questão são apresentadas as principais vantagens e desvantagens do método:

a) Vantagens do FDTD:

- permite a avaliação de transitórios, pois todos os cálculos são realizados no domínio tempo. Métodos que utilizam fasores são aplicáveis apenas para regime permanente;
- além da resposta no tempo, também permite a obtenção da resposta no domínio da frequência, a partir da transformada de Fourier da resposta no tempo para regime permanente;
- arranjos tridimensionais de geometria complexa e com vários materiais de diferentes propriedades eletromagnéticas (desde condutores até dielétricos) são facilmente tratados, com o mesmo modelo matemático;
- não depende da função de excitação das fontes. Fasores são aplicáveis apenas para estímulos senoidais. Portanto, para utilização de métodos como MoM é necessário decompor o sinal não-senoidal em diversas senóides, realizar a simulação do sistema para cada componente e, por último, compor a resposta final por superposição;
- resolve simultaneamente as equações de campo elétrico e magnético no processo de cálculo, ao contrário de outros métodos. Ou seja, tanto as propriedade elétricas quanto magnéticas do meio são consideradas, o que torna o algoritmo mais robusto [1];
- simula sistemas de difícil modelamento por outros métodos (MoM). Entre as aplicações pode-se citar: emissão em placas de circuito impresso, radar, absorção de energia pelo corpo humano, etc.

b) Desvantagens:

- exigência computacional: método necessita grande área de memória RAM e velocidade de processamento elevada, o que até pouco tempo impedia sua implementação em microcomputadores (PCs de uso geral);

- como o *grid* tem tamanho limitado, existe uma descontinuidade nas fronteiras (bordas) do *grid*. Ao alcançar as bordas, a onda sofre reflexão numérica pela descontinuidade existente, o que torna o método inútil se não houver o tratamento adequado do problema;
- a simulação de materiais cuja constante dielétrica depende da frequência não é realizada de forma direta, pois o processo de cálculo é realizado no domínio tempo, não permitindo avaliar a variação da propriedade.

Ao longo do trabalho de integração a ser desenvolvido, pretende-se mostrar que as limitações de exigência computacional e reflexão numérica não precisam ser mais consideradas, pois já tiveram solução.

## 4. Fundamentação Teórica

O grande problema de resolver simultaneamente as duas equações de Maxwell é o fato de a solução de uma equação implica necessariamente na solução prévia da outra equação associada.

Para melhor compreender o problema, vamos repetir as equações diferenciais de Maxwell [1][2], considerando os vetores  $\mathbf{J}_{source}$  e  $\mathbf{M}_{source}$  nulos:

$$\frac{\partial \mathbf{H}}{\partial t} = \frac{1}{\mu} (-\nabla \times \mathbf{E} - \sigma^* \mathbf{H}) \quad (\text{Eq.5})$$

$$\frac{\partial \mathbf{E}}{\partial t} = \frac{1}{\varepsilon} (\nabla \times \mathbf{H} - \sigma \mathbf{E}) \quad (\text{Eq.6})$$

As quais, a partir da representação matricial dos vetores, tomam as seguintes formas:

$$\frac{\partial}{\partial t} \begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} = -\frac{1}{\mu} \begin{bmatrix} \frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z} \\ \frac{\partial E_x}{\partial z} - \frac{\partial E_z}{\partial x} \\ \frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y} \end{bmatrix} - \frac{\sigma^*}{\mu} \begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} \quad (\text{Eq.5a})$$

$$\frac{\partial}{\partial t} \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} = \frac{1}{\varepsilon} \begin{bmatrix} \frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} \\ \frac{\partial H_x}{\partial z} - \frac{\partial H_z}{\partial x} \\ \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} \end{bmatrix} - \frac{\sigma}{\varepsilon} \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} \quad (\text{Eq.6a})$$

Nota-se que para determinar a variação campo magnético no tempo é preciso ser conhecida a variação do campo elétrico no espaço. E a recíproca é verdadeira para o campo elétrico.

Yee resolveu o problema de forma muito elegante, introduzindo um artifício numérico que permite a solução isolada das duas equações.

## 4.1 O Algoritmo de Yee

Como referido na introdução ao método FDTD, deve ser realizada a avaliação dos campos elétricos e magnéticos em todas as células do *grid* (varredura), para então ser possível avançar no tempo e realizar nova varredura. Ou seja, a cada intervalo  $\Delta t$  ocorre uma varredura do *grid*.

Para a solução “simultânea” das equações de Maxwell de forma numérica, Kane Yee propôs dividir o intervalo  $\Delta t$  em dois e realizar a varredura do *grid* em duas etapas:

- a) Calcular o valor atual  $\mathbf{E}$  a partir do valor passado  $\mathbf{E}$  na célula e dos valores passados  $\mathbf{H}$  das células vizinhas.
- b) Calcular o valor atual  $\mathbf{H}$  a partir do valor passado de  $\mathbf{H}$  na célula e dos valores atuais  $\mathbf{E}$  das células vizinhas.

Isso significa dizer que: primeiro ocorre a alteração em um dos campos (no caso:  $\mathbf{E}$ ), que após provoca a alteração no outro (no caso:  $\mathbf{H}$ ), que por sua vez altera novamente o primeiro e assim sucessivamente, ao longo do tempo e espaço.

Supondo que o intervalo de tempo  $\Delta t$  é infinitesimal (e mais ainda é  $\Delta t/2$ ), pode-se considerar que os dois fenômenos ocorrem ao simultaneamente. E, ao observar-se atentamente as equações, percebe-se que era essa a visão de Maxwell.

Ainda, para implementar sua idéia, Kane Yee apresentou um arranjo especial dos campos no espaço, a fim de resolver as equações de Maxwell utilizando a aproximação numérica pelo método da diferenças finitas centrais.

Como será apresentado no desenvolvimento matemático, o arranjo proposto por Yee permite a avaliação das diferenciais parciais do problema através de diferenças finitas, a partir da expansão de Taylor.

O método apresentado por Yee pode ser desenvolvido em qualquer sistema de coordenadas, embora tenha sido originalmente apresentado no sistema cartesiano, utilizando-se como simplificação células de formato cúbico.

Ainda hoje, as idéias originais de Yee para o arranjo das componentes do campo elétrico e magnético no espaço, as quais são ilustradas na figura a seguir:

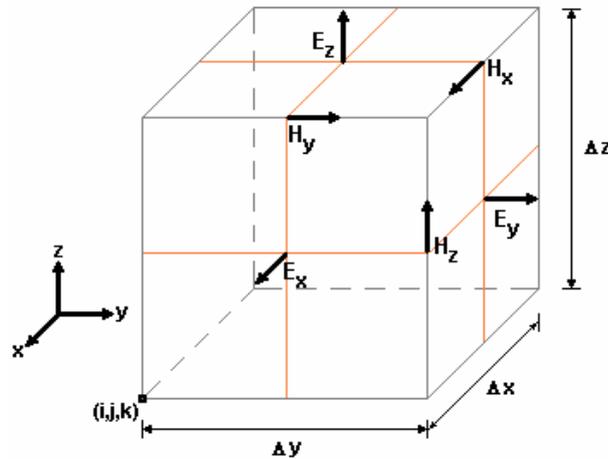


Fig. 1 – Disposição dos componentes dos vetores  $\mathbf{E}$  e  $\mathbf{H}$  em uma célula cúbica [4]

O arranjo acima apresentado foi denominado: *leapfrog arrangement* e o processo iterativo de cálculo: *leapfrog time-stepping*. Para facilitar o entendimento, a seguinte notação é adotada:

$$\mathbf{U}(x_i, y_j, z_k, t_n) = \mathbf{U}(i \cdot \Delta x, j \cdot \Delta y, k \cdot \Delta z, n \cdot \Delta t) = \mathbf{U}(i, j, k, n) = \mathbf{U}_{i,j,k}^n$$

onde:

$\mathbf{U}$  – vetor de campo elétrico ou magnético

$\Delta x, \Delta y, \Delta z$  – dimensões da célula

$\Delta t$  – intervalo de tempo entre varreduras

$i, j, k$  – índice que localizam a célula no *grid*

$n$  – índice de tempo

Assim, cada célula do *grid* é definida por um ponto de referência  $(x_i, y_j, z_k)$  e por suas dimensões  $\Delta x, \Delta y$  e  $\Delta z$ , ocupando o espaço limitado pelas coordenadas:

Tab. 1 Coordenadas limítrofes de uma célula

Posição	Notação simplificada
$(x_i, y_j, z_k)$	$(i, j, k)$
$(x_i, y_{j+\Delta y}, z_k)$	$(i, j+1, k)$
$(x_{i-\Delta x}, y_j, z_k)$	$(i-1, j, k)$
$(x_{i-\Delta x}, y_{j+\Delta y}, z_k)$	$(i-1, j+1, k)$
$(x_i, y_j, z_{k+\Delta z})$	$(i, j, k+1)$
$(x_{i-\Delta x}, y_j, z_{k+\Delta z})$	$(i-1, j, k+1)$
$(x_{i-\Delta x}, y_{j+\Delta y}, z_{k+\Delta z})$	$(i-1, j+1, k+1)$
$(x_i, y_{j+\Delta y}, z_{k+\Delta z})$	$(i, j+1, k+1)$

E as componentes dos vetores associados àquela célula referenciada pelo ponto (i, j, k) estão localizadas nas seguintes coordenadas:

Tab. 2 – Coordenadas das componentes de campo associadas a uma célula

Componente	Posição (notação simplificada)
$E_x$	( i, j+1/2, k+1/2 )
$E_y$	( i-1/2, j+1, k+1/2 )
$E_z$	( i-1/2, j+1/2, k+1 )
$H_x$	( i-1/2, j+1, k+1 )
$H_y$	( i, j+1/2, k+1 )
$H_z$	( i, j+1, k+1/2 )

Originalmente Yee apresentou seu método, considerando meios sem perdas ( $\sigma=0$  e  $\sigma^*=0$ ). Porém, o tratamento matemático apresentado na seqüência considera meios reais.

## 4.2 Desenvolvimento matemático

Utilizando as componentes de campo, as duas equações de Maxwell na forma vetorial são desdobradas em seis equações diferenciais escalares:

$$\frac{\partial H_x}{\partial t} = \frac{1}{\mu} \left( \frac{\partial E_y}{\partial z} - \frac{\partial E_z}{\partial y} - \sigma^* H_x \right) \quad (\text{Eq.7})$$

$$\frac{\partial H_y}{\partial t} = \frac{1}{\mu} \left( \frac{\partial E_z}{\partial x} - \frac{\partial E_x}{\partial z} - \sigma^* H_y \right) \quad (\text{Eq.8})$$

$$\frac{\partial H_z}{\partial t} = \frac{1}{\mu} \left( \frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x} - \sigma^* H_z \right) \quad (\text{Eq.9})$$

$$\frac{\partial E_x}{\partial t} = \frac{1}{\varepsilon} \left( \frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} - \sigma E_x \right) \quad (\text{Eq.10})$$

$$\frac{\partial E_y}{\partial t} = \frac{1}{\varepsilon} \left( \frac{\partial H_x}{\partial z} - \frac{\partial H_z}{\partial x} - \sigma E_y \right) \quad (\text{Eq.11})$$

$$\frac{\partial E_z}{\partial t} = \frac{1}{\varepsilon} \left( \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} - \sigma E_z \right) \quad (\text{Eq.12})$$

As derivadas parciais foram resolvidas em termos de diferenças finitas, com o emprego da expansão de Taylor. A seguir é apresentada a solução, considerando

uma variável de duas dimensões, a qual é análoga se a variável possuir quatro dimensões (três no espaço e uma no tempo).

A expansão, utilizando a série de Taylor, de uma função  $U(x, t)$  em torno do ponto  $x_i$  e  $x_i + \frac{\Delta x}{2}$  em dado tempo  $t_n$  é dada pela equação abaixo:

$$U\left(x_i + \frac{\Delta x}{2}, t_n\right) = U(x_i, t_n) + \frac{\Delta x}{2} \cdot \frac{\partial U(x_i, t_n)}{\partial x} + \frac{\left(\frac{\Delta x}{2}\right)^2}{2} \cdot \frac{\partial^2 U(x_i, t_n)}{\partial x^2} + \frac{\left(\frac{\Delta x}{2}\right)^3}{6} \cdot \frac{\partial^3 U(x_i, t_n)}{\partial x^3} + \frac{\left(\frac{\Delta x}{2}\right)^4}{24} \cdot \frac{\partial^4 U(\xi_1, t_n)}{\partial x^4} \quad (\text{Eq.13})$$

onde:  $\xi_1$  – é um ponto no espaço entre ponto  $x_i$  e  $x_i + \frac{\Delta x}{2}$

Da mesma forma, em torno dos pontos ponto  $x_i$  e  $x_i - \frac{\Delta x}{2}$  tem-se:

$$U\left(x_i - \frac{\Delta x}{2}, t_n\right) = U(x_i, t_n) - \frac{\Delta x}{2} \cdot \frac{\partial U(x_i, t_n)}{\partial x} + \frac{\left(\frac{\Delta x}{2}\right)^2}{2} \cdot \frac{\partial^2 U(x_i, t_n)}{\partial x^2} - \frac{\left(\frac{\Delta x}{2}\right)^3}{6} \cdot \frac{\partial^3 U(x_i, t_n)}{\partial x^3} + \frac{\left(\frac{\Delta x}{2}\right)^4}{24} \cdot \frac{\partial^4 U(\xi_2, t_n)}{\partial x^4} \quad (\text{Eq.14})$$

onde:  $\xi_2$  – é um ponto no espaço entre  $x_i$  e  $x_i - \frac{\Delta x}{2}$

Assim, na subtração (Eq.13) e (Eq.14), tem-se:

$$U\left(x_i + \frac{\Delta x}{2}, t_n\right) - U\left(x_i - \frac{\Delta x}{2}, t_n\right) = \Delta x \cdot \frac{\partial U(x_i, t_n)}{\partial x} + O[(\Delta x)^2] \quad (\text{Eq.15})$$

onde:

$O[(\Delta x)^2]$  – termo residual, proporcional ao quadro do incremento  $\Delta x$  e que representa as derivadas de ordem superior.

Supondo o incremento  $\Delta x$  suficientemente pequeno, pode-se desprezar o termo residual e a equação (Eq.15) rearranjada tem a seguinte forma:

$$\frac{\partial U(x_i, t_n)}{\partial x} = \frac{U\left(x_i + \frac{\Delta x}{2}, t_n\right) - U\left(x_i - \frac{\Delta x}{2}, t_n\right)}{\Delta x} \quad (\text{Eq.16})$$

Ou, na notação simplificada:

$$\frac{\partial U(i, n)}{\partial x} = \frac{U_{i+1/2}^n - U_{i-1/2}^n}{\Delta x} \quad (\text{Eq.17})$$

De forma análoga, pode-se demonstrar que a derivada parcial com relação ao tempo é dada por:

$$\frac{\partial U(i,n)}{\partial t} = \frac{U_i^{n+1/2} - U_i^{n-1/2}}{\Delta t} \quad (\text{Eq.18})$$

Assim sendo, considerando ainda:

- a célula cúbica ( $\Delta x = \Delta y = \Delta z = \Delta$ );
- o campo elétrico  $\mathbf{E}$  é processado primeiro (ou seja, em  $\Delta t/2$ ).
- $U(i,n) = \frac{U_i^{n+1/2} + U_i^{n-1/2}}{2}$  é uma aproximação válida e precisa

E após a substituição das derivadas parciais pelas respectivas relações de diferenças e rearranjo adequado dos termos, as equações diferenciais de Maxwell tomam a seguinte forma:

$$E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+1/2} = Ca_{i,j+\frac{1}{2},k+\frac{1}{2}} \cdot E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n-1/2} + Cb_{i,j+\frac{1}{2},k+\frac{1}{2}} \left( \begin{array}{l} H_z \Big|_{i,j+1,k+\frac{1}{2}}^n - H_z \Big|_{i,j,k+\frac{1}{2}}^n + \\ H_y \Big|_{i,j+\frac{1}{2},k}^n - H_y \Big|_{i,j+\frac{1}{2},k+1}^n \end{array} \right) \quad (\text{Eq.19})$$

$$E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+1/2} = Ca_{i-\frac{1}{2},j+1,k+\frac{1}{2}} \cdot E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n-1/2} + Cb_{i-\frac{1}{2},j+1,k+\frac{1}{2}} \left( \begin{array}{l} H_x \Big|_{i-\frac{1}{2},j+1,k+1}^n - H_x \Big|_{i-\frac{1}{2},j+1,k}^n + \\ H_z \Big|_{i-1,j+1,k+\frac{1}{2}}^n - H_z \Big|_{i,j+1,k+\frac{1}{2}}^n \end{array} \right) \quad (\text{Eq.20})$$

$$E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+1/2} = Ca_{i-\frac{1}{2},j+\frac{1}{2},k+1} \cdot E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n-1/2} + Cb_{i-\frac{1}{2},j+\frac{1}{2},k+1} \left( \begin{array}{l} H_y \Big|_{i,j+\frac{1}{2},k+1}^n - H_y \Big|_{i-1,j+\frac{1}{2},k+1}^n + \\ H_x \Big|_{i-\frac{1}{2},j,k+1}^n - H_x \Big|_{i-\frac{1}{2},j+1,k+1}^n \end{array} \right) \quad (\text{Eq.21})$$

$$H_x \Big|_{i-\frac{1}{2},j+1,k+1}^{n+1} = Da_{i-\frac{1}{2},j+1,k+1} \cdot H_x \Big|_{i-\frac{1}{2},j+1,k+1}^n + Db_{i-\frac{1}{2},j+1,k+1} \left( \begin{array}{l} E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{3}{2}}^{n+\frac{1}{2}} - E_y \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} + \\ E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} - E_z \Big|_{i-\frac{1}{2},j+\frac{3}{2},k+1}^{n+\frac{1}{2}} \end{array} \right) \quad (\text{Eq.22})$$

$$H_y \Big|_{i,j+\frac{1}{2},k+1}^{n+1} = Da_{i,j+\frac{1}{2},k+1} \cdot H_y \Big|_{i,j+\frac{1}{2},k+1}^n + Db_{i,j+\frac{1}{2},k+1} \left( \begin{array}{c} E_z \Big|_{i+\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} - E_z \Big|_{i-\frac{1}{2},j+\frac{1}{2},k+1}^{n+\frac{1}{2}} + \\ E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} - E_x \Big|_{i,j+\frac{1}{2},k+\frac{3}{2}}^{n+\frac{1}{2}} \end{array} \right) \quad (\text{Eq.23})$$

$$H_z \Big|_{i,j+1,k+\frac{1}{2}}^{n+1} = Da_{i,j+1,k+\frac{1}{2}} \cdot H_z \Big|_{i,j+1,k+\frac{1}{2}}^n + Db_{i,j+1,k+\frac{1}{2}} \left( \begin{array}{c} E_x \Big|_{i,j+\frac{3}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} - E_x \Big|_{i,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} + \\ E_z \Big|_{i-\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} - E_z \Big|_{i+\frac{1}{2},j+1,k+\frac{1}{2}}^{n+\frac{1}{2}} \end{array} \right) \quad (\text{Eq.24})$$

Nota-se no índice de tempo a aplicação do conceito *leapfrog time-stepping*.

Ainda, os coeficientes  $Ca$ ,  $Cb$ ,  $Da$ ,  $Db$  dependem apenas das propriedades elétricas e magnéticas do material da célula, da dimensão  $\Delta$  da célula cúbica e do incremento de tempo  $\Delta t$ , cujas expressões são apresentadas abaixo:

$$Ca_{i,j,k} = \frac{1 - \frac{\sigma_{i,j,k} \cdot \Delta t}{2\epsilon_{i,j,k}}}{1 + \frac{\sigma_{i,j,k} \cdot \Delta t}{2\epsilon_{i,j,k}}} \quad (\text{Eq.25})$$

$$Cb_{i,j,k} = \frac{\frac{\Delta t}{\Delta \cdot \epsilon_{i,j,k}}}{1 + \frac{\sigma_{i,j,k} \cdot \Delta t}{2\epsilon_{i,j,k}}} \quad (\text{Eq.26})$$

$$Da_{i,j,k} = \frac{1 - \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}}{1 + \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}} \quad (\text{Eq.27})$$

$$Db_{i,j,k} = \frac{\frac{\Delta t}{\Delta \cdot \mu_{i,j,k}}}{1 + \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}} \quad (\text{Eq.28})$$

A demonstração matemática das equações acima é apresentada em detalhes nas referências pesquisadas.

### **4.3 Tamanho de célula e incremento de tempo**

Conforme apresentado na introdução ao método, tanto a dimensão  $\Delta$  da célula cúbica quanto o incremento  $\Delta t$  de tempo são função do comprimento de onda  $\lambda$  da eletromagnética em propagação.

Taflov [1], nos capítulos 3 e 4 de seu livro, avalia em detalhes os valores necessários para assegurar a estabilidade numérica, a convergência e a obtenção de resultados adequados. A seguir, nos limitamos a apresentar as relações “mágicas”, através da definição de dois parâmetros:

$$N = \frac{\lambda}{\Delta} \quad (\text{Eq.29})$$

$$S = \frac{c \cdot \Delta t}{\Delta} \quad (\text{Eq.30})$$

onde:  $c$  – velocidade da luz no vácuo

Os valores críticos para os parâmetros são:  $N_{\text{mín}}=10$  e  $S_{\text{máx}}=\frac{1}{\sqrt{3}}$ . E os valores normalmente adotados são:  $N \geq 10$  e  $S=0,5$ .

### **4.4 Fontes de excitação**

Até o momento as fontes externas de campo foram consideradas nulas, porém sua inserção no modelo é extremamente fácil. As fontes são classificadas em dois tipos distintos, conforme sua natureza:

- a) *hard sources* – o valor do campo da célula na qual é aplicada a fonte é igual ao valor da fonte e independente da interação com as células vizinhas.
- b) *soft source* – o valor do campo da célula na qual é aplicada a fonte é igual a adição do valor da fonte e do valor de campo resultados da sua interação com as células vizinhas.

Para exemplificar a situação, vamos considerar a propagação de uma onda elétrica apenas na direção do eixo de coordenadas  $x$ . Ainda, na célula na posição  $i_s$  existe uma fonte de campo elétrico com valor determinado pela função  $f(t)$ . Então, a única componente de campo elétrico de interesse  $E_z$  é definida conforme o tipo de fonte existente:

- *hard source*:

$$E_z \Big|_{i_s}^{n+1/2} = f(t) = f((n + \frac{1}{2}) \cdot \Delta t) \quad (\text{Eq.31})$$

- *soft source*:

$$E_z \Big|_{i_s}^{n+1/2} = E_z \Big|_{i_s}^{n-1/2} + \frac{\Delta t}{\epsilon \cdot \Delta} (H_y \Big|_{i_s+1/2}^n - H_y \Big|_{i_s-1/2}^n) + f((n + \frac{1}{2}) \cdot \Delta t) \quad (\text{Eq.32})$$

Na verdade, conforme apresentado em [2], pp. 517-519, a definição de uma fonte de campo elétrico ou magnético como uma *hard source* não representa adequadamente a situação física para uma fonte transitória (pulsos), uma vez que não são considerados os outros fenômenos na célula da fonte após a fonte deixar de existir (o valor nulo permanece forçado na célula).

De forma mais clara, em uma célula na qual se impõe uma *hard source*, o valor de campo associado é determinado exclusivamente pela fonte, não sendo afetado pelo valor de campo das células vizinhas, conforme preconiza as equações de Maxwell, mesmo após a fonte tenha deixado de existir (fontes transitórias). Isto se caracteriza por uma singularidade local, que resulta em reflexão numérica.

A idéia de realizar o modelamento das fontes como *soft sources* é justamente eliminar a natureza reflexiva do outro tipo de fonte, permitindo uma representação realística do fenômeno físico. Contudo, da forma como a *soft source* está modelada, a célula da fonte não tem seu valor definido somente pela fonte. Isso é no mínimo estranho, pois conceitualmente a existência de uma fonte não nula deve definir o valor local da propriedade (tensão, corrente, campo, etc.).

Portanto, quando presentes no *grid*, as fontes serão modeladas como *hard sources* enquanto não nulas. Senão, serão aplicadas as equações de Maxwell às células das fontes nulas (a não ser que a célula em questão seja especificado como um ponto de aterramento).

## 4.5 Equacionamento para cálculo numérico

As equações (Eq.19) a (Eq.32) formam a base da ferramenta de simulação desenvolvida. Contudo, para ser possível sua utilização em um algoritmo numérico, são necessárias algumas considerações e adaptações, conforme descrito a seguir.

Cada célula é identificada pela posição do seu ponto de referência na notação simplificada, definida genericamente por um índice de três argumentos (i, j, k), cada qual associado respectivamente aos eixos cartesianos X, Y e Z, conforme já apresentado. Portanto, ao percorrer o *grid*, cada incremento unitário em um dos argumentos do índice corresponde ao deslocamento de uma célula e, obviamente, a distância equivalente à  $\Delta$ .

As componentes de campo associadas a cada célula serão identificadas pelo próprio índice da célula. Assim, na célula referenciada por (i, j, k) os vetores associados são:

Tab. 3 – Identificação das componentes de campo

Componente	Posição (notação simplificada)	Nova referência
$E_x$	( i, j+1/2, k+1/2 )	(i, j, k)
$E_y$	( i-1/2, j+1, k+1/2 )	(i, j, k)
$E_z$	( i-1/2, j+1/2, k+1 )	(i, j, k)
$H_x$	( i-1/2, j+1, k+1 )	(i, j, k)
$H_y$	( i, j+1/2, k+1 )	(i, j, k)
$H_z$	( i, j+1, k+1/2 )	(i, j, k)

A evolução no tempo do sistema é identificada pelo índice n, sendo cada incremento unitário de n correspondente ao intervalo de tempo  $\Delta t$ , anteriormente definido na fundamentação teórica.

Para cada intervalo de tempo, todos os valores de campo são recalculados. Seguindo o procedimento estabelecido por Yee, em primeiro lugar são processadas as componentes de campo elétrico de todo o *grid*.

Após, com os valores atualizados de campo elétrico, são processadas as componentes de campo magnético. Assim, para o instante n as equações são reescritas da seguinte forma:

$$E_x|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot E_x|_{i,j,k}^n + Cb_{i,j,k} \left( H_z|_{i,j,k}^n - H_z|_{i,j-1,k}^n + H_y|_{i,j,k-1}^n - H_y|_{i,j,k}^n \right) \quad (\text{Eq.33})$$

$$E_y|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot E_y|_{i,j,k}^n + Cb_{i,j,k} \left( H_x|_{i,j,k}^n - H_x|_{i,j,k-1}^n + H_z|_{i-1,j,k}^n - H_z|_{i,j,k}^n \right) \quad (\text{Eq.34})$$

$$E_z|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot E_z|_{i,j,k}^n + Cb_{i,j,k} \left( H_y|_{i,j,k}^n - H_y|_{i-1,j,k}^n + H_x|_{i,j-1,k}^n - H_x|_{i,j,k}^n \right) \quad (\text{Eq.35})$$

$$H_x|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot H_x|_{i,j,k}^n + Db_{i,j,k} \left( E_y|_{i,j,k+1}^{n+1} - E_y|_{i,j,k}^{n+1} + E_z|_{i,j,k}^{n+1} - E_z|_{i,j+1,k}^{n+1} \right) \quad (\text{Eq.36})$$

$$H_y|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot H_y|_{i,j,k}^n + Db_{i,j,k} \left( E_z|_{i+1,j,k}^{n+1} - E_z|_{i,j,k}^{n+1} + E_x|_{i,j,k}^{n+1} - E_x|_{i,j,k+1}^{n+1} \right) \quad (\text{Eq.37})$$

$$H_z|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot H_z|_{i,j,k}^n + Db_{i,j,k} \left( E_x|_{i,j+1,k}^{n+1} - E_x|_{i,j,k}^{n+1} + E_y|_{i,j,k}^{n+1} - E_y|_{i+1,j,k}^{n+1} \right) \quad (\text{Eq.38})$$

E, na forma apresentada, torna-se possível implementar a rotina responsável pelo cálculo dos valores de campo nas sucessivas interações de tempo.

#### 4.6 ABC (Absorbing Boundary Conditions)

Relembrando a fundamentação teórica, devido ao *grid* ter tamanho limitado, existe descontinuidade nas fronteiras (bordas) do *grid*. Ao alcançar as bordas, a onda sofre reflexão numérica pela descontinuidade existente, o que torna o método inútil se não houver o tratamento adequado do problema.

Uma rápida inspeção nas equações (Eq.33) a (Eq.38), percebe-se que as seguintes componentes não estão definidas:

Tab. 4 – Bordas do *grid* e as componentes de campo não definidas

Borda	Componentes não definidas
$i = 0$	$H_y(-1, j, k, n)$ e $H_z(-1, j, k, n)$
$j = 0$	$H_z(i, -1, k, n)$ e $H_x(i, -1, k, n)$
$k = 0$	$H_x(i, j, -1, n)$ e $H_y(i, j, -1, n)$
$i = i_{\max}$	$E_y(i_{\max}+1, j, k, n)$ e $E_z(i_{\max}+1, j, k, n)$
$j = j_{\max}$	$E_x(i, j_{\max}+1, k, n)$ e $E_z(i, j_{\max}+1, k, n)$
$k = k_{\max}$	$E_y(i, j, k_{\max}+1, n)$ e $E_z(i, j, k_{\max}+1, n)$

Desde a apresentação do algoritmo de Yee, muitos pesquisadores dedicaram seus esforços para encontrar uma forma adequada de realizar o truncamento do *grid*, ou seja, como determinar os valores adequados para as componentes de campo faltantes nas bordas.

A primeira forma possível de contornar esse problema é fazer o *grid* tão grande quanto necessário para evitar que a onda eletromagnética atinja as bordas do *grid* ao longo das interações. Assim, considerando que todos os componentes de campo são inicialmente nulos, pode-se atribuir esse mesmo valor nulo às componentes de campo não definidas (ver Tab. 6). Mantido esse valor nulo para essas componentes ao longo de todo o processo de cálculo, não haverá erro numérico caso a onda não alcance as bordas.

Mas isso pode tornar a solução computacional inviável, pelo tamanho de memória necessário para acomodar o *grid* e todas as variáveis associadas.

O *grid* poderia ser subdividido (*subgrid*), os quais seriam processados em computadores separados, os quais trocariam informações entre si para transferência das variáveis correspondentes as fronteiras de cada *subgrid*.

Porém, esbarra-se em um problema: “Qual o tamanho necessário do *grid* ?” A princípio, parece uma pergunta fácil de responder. Tendo-se definidos os incrementos  $\Delta$  e  $\Delta t$ , é possível determinar o número de células necessárias para, além de conter todos os objetos e fontes do sistema, compor um *grid* grande o suficiente para que nenhuma onda do sistema alcançasse suas as bordas durante o tempo previsto de processamento (isto é, desde o início até o tempo final que, quando atingido, o processo de interações é interrompido).

Contudo, surge uma nova pergunta: “Qual será o tempo final suficiente para o sistema modelado atingir o regime ou, pelo menos, para que todas as perturbações significativas ficaram evidentes para que a simulação tenha valor?”

Ainda, existem sistemas para os quais se deseja simular a incidência uma frente de onda sobre os objetos contidos em determinada região (é o caso em que a fonte está distante dos objetos irradiados). Esse é um problema de engenharia bastante comum, que implica em a frente de onda ter dimensões necessariamente muito maiores que o conjunto de objetos presentes. Como conseqüência, seria preciso um número absurdamente grande de células para conciliar um incremento espacial  $\Delta$  suficientemente pequeno para caracterizar os objetos contidos no sistema

e a dimensão total do *grid* necessária para modelar uma frente de onda de forma adequada.

Por último, além de implementar o algoritmo de simulação, o software deveria ser desenvolvido para realizar o ajuste dinâmico da estrutura computacional a ser utilizada (número de computadores, alocação de memória, transferência de valores, etc.), o que poderia se tornar um desafio maior que a implementação do algoritmo de simulação em si, fugindo aos objetivos do trabalho.

A partir dessas considerações, percebeu-se inevitável a busca por outro método para contornar o problema numérico associados às fronteiras do *grid*. Algumas alternativas são apresentadas a seguir.

#### **4.6.1 Métodos Analíticos**

As primeiras tentativas de solução para o problema de reflexão nas bordas buscaram uma solução analítica, a partir de outras equações acessórias que, aplicadas às bordas, pudessem minimizar a reflexão existente.

A maioria dos métodos elaborados na época (Taflov [1], capítulo 6) parte da equação que descreve o deslocamento de uma onda no espaço, apresentada abaixo:

$$\frac{\partial^2 U}{\partial t^2} = c^2 \nabla^2 U \quad (\text{Eq.39})$$

onde:

$U$  – componente do vetor de campo elétrico ou magnético

$c$  – velocidade de propagação da luz no vácuo

A Equação (Eq.39), em coordenadas cartesianas, pode ser reescrita da seguinte forma:

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} - \frac{1}{c^2} \cdot \frac{\partial^2 U}{\partial t^2} = 0 \quad (\text{Eq.40})$$

A partir da equação escalar foram desenvolvidos operadores que conduzem a uma expressão para estimação de cada componente nas bordas de interesse.

Provavelmente, a solução mais citada foi a desenvolvida por Mur, a partir do operador apresentado por Engquist and Majda.

A técnica proposta por Mur permitiu que os erros de reflexão numérica fossem aceitáveis para sua utilização.

Porém, as técnicas baseadas no método analítico, a partir da equação da onda, têm alguns inconvenientes (Taflov [1], capítulo 6):

- a) é considerado que a onda se desloca a velocidade da luz junto a borda, independente do meio existente e da faixa de freqüência em avaliação;
- b) existe a dependência com o ângulo de incidência da onda junto as bordas, apresentando erros sensivelmente maiores, quanto maior o ângulo de incidência em relação ao plano (borda) em estudo;
- c) erro por reflexão é elevado em comparação com outras técnicas.

#### **4.6.2 Métodos Numéricos**

A idéia de buscar avaliar o valor da componente de campo faltante em função de valores conhecidos da componente, utilizando métodos de cálculo numérico, foi proposta pela primeira vez por Liao et al [5].

De forma resumida, a proposta de Liao consiste em estimar o próximo valor da componente faltante em uma das bordas, a partir dos valores já calculados da componente em células do *grid*, na direção perpendicular a borda. Ainda, a fim de levar em conta o atraso existente na propagação da onda a cada interação, os valores são tomados progressivamente retardados de  $\Delta t$ . O conceito fica mais evidente quando apresentado o seu equacionamento.

Seja a borda de interesse perpendicular ao eixo X, localizada em  $x_{\max}$  e sejam:

$$U_0 = U(x_{\max}, t + \Delta t), \text{ o valor desejado}$$

$$U_1 = U(x_{\max} - h, t), \text{ conhecido}$$

$$U_2 = U(x_{\max} - 2h, t - \Delta t), \text{ conhecido}$$

$$U_3 = U(x_{\max} - 3h, t - 2\Delta t), \text{ conhecido}$$

$$U_L = U(x_{\max} - Lh, t - (L-1)\Delta t), \text{ conhecido}$$

onde:

$U$  – componente do vetor de campo elétrico ou magnético

$x_{\max}$  – a localização da componente desejada;

1, 2, 3, ..., L – pontos nos quais a componente é conhecida;

L – número de pontos nos quais a componente é conhecida;

$h = \alpha \cdot c \cdot \Delta t$  ( $0,5 \leq \alpha \leq 2$ ) – o espaçamento entre os pontos

Então, valor desejado da componente pode ser estimado utilizando uma adaptação do método de interpolação por diferenças finitas de Newton. Na verdade, os conceitos originalmente de Liao são desenvolvidos de forma mais complexa. Mas essa a abordagem é adotada por Taflov [1], capítulo 6, para apresentar as idéias de Liao em um formato de mais fácil compreensão.

Segundo o método de interpolação por diferenças finitas de Newton, uma função, cujos valores são conhecidos em L pontos igualmente espaçados por h, pode ter seu valor estimado no ponto de interesse  $x = x_{\max} - A \cdot h$  pela expressão:

$$U_A = U_1 + \frac{\beta \cdot \Delta^1 U_1}{1!} + \frac{\beta(\beta+1) \cdot \Delta^2 U_1}{2!} + \frac{\beta(\beta+1)(\beta+2) \cdot \Delta^3 U_1}{3!} + \dots + \frac{\beta(\beta+1)(\beta+2) \dots (\beta+L-2) \cdot \Delta^{L-1} U_1}{(L-1)!} \quad (\text{Eq.41})$$

onde:

$1 \leq A \leq L$ , por definição

$\beta = 1 - A$

$\Delta^{L-1} U_1$  – diferença finita de ordem L-1 em relação ao ponto 1

As diferenças finitas são calculadas conforme ilustrado na tabela abaixo.

Tab. 5 – Cálculo das diferenças finitas

ordem 1	ordem 2	...	ordem L-2	ordem L-1
$\Delta^1 U_1 = U_1 - U_2$	$\Delta^2 U_1 = \Delta^1 U_1 - \Delta^1 U_2$	...	$\Delta^{L-2} U_1 = \Delta^{L-3} U_1 - \Delta^{L-3} U_2$	$\Delta^{L-1} U_1 = \Delta^{L-2} U_1 - \Delta^{L-2} U_2$
$\Delta^1 U_2 = U_2 - U_3$	$\Delta^2 U_2 = \Delta^1 U_2 - \Delta^1 U_3$	...	$\Delta^{L-2} U_2 = \Delta^{L-3} U_2 - \Delta^{L-3} U_3$	X
$\Delta^1 U_3 = U_3 - U_4$	$\Delta^2 U_3 = \Delta^1 U_3 - \Delta^1 U_4$	...	X	X
...	...	...	...	...
$\Delta^1 U_{L-2} = U_{L-2} - U_{L-1}$	$\Delta^1 U_{L-2} = \Delta^1 U_{L-2} - \Delta^1 U_{L-1}$	X	X	X
$\Delta^1 U_{L-1} = U_{L-1} - U_L$	X	X	X	X

Contudo, apesar do método de Newton ter sido desenvolvido para a interpolação, a idéia de Liao pode ser interpretada pelo uso do método de Newton para extrapolar o valor da componente  $U_0$ . Nessa hipótese, tem-se  $A = 0$  e  $\beta = 1$ , o que simplifica a equação (Eq.41) conforme segue:

$$U_0 = U_1 + \Delta^1 U_1 + \Delta^2 U_1 + \Delta^3 U_1 + \dots + \Delta^{L-1} U_1 \quad (\text{Eq.42})$$

Ainda, conforme Taflov [1], capítulo 6, é possível alcançar erros inferiores a 0,5% em simulações tridimensionais, para  $N=20$  e  $S=0,5$  (definidos por Eq. 29 e Eq. 30), adotando  $L = 3$ , se usada representação em ponto flutuante de precisão dupla.

Além disso, é possível escolher  $\alpha = 1/S = 2$  para que se tenha  $h = \Delta$ , permitindo a utilização direta dos valores calculados das componentes no *grid*. Dessa forma, a equação (Eq.42) fica reduzida para a forma abaixo:

$$U_0 = U_1 + \Delta^1 U_1 + \Delta^2 U_1 = 3U_1 - 3U_2 + U_3 \quad (\text{Eq.43})$$

onde:

$$U_1 = U(x_{\max} - \Delta, t)$$

$$U_2 = U(x_{\max} - 2\Delta, t - \Delta t)$$

$$U_3 = U(x_{\max} - 3\Delta, t - 2\Delta t)$$

Os conceitos apresentados acima podem ser aplicados a todas as bordas. As expressões para o cálculo das componentes faltantes, conforme a abordagem de Liao e adaptadas à simbologia adotada, são apresentadas na seqüência.

Em  $i = 0$ :

$$H_y \Big|_{-1,j,k}^n = 3H_y \Big|_{0,j,k}^n - 3H_y \Big|_{1,j,k}^{n-1} + H_y \Big|_{2,j,k}^{n-2} \quad (\text{Eq.44})$$

$$H_z \Big|_{-1,j,k}^n = 3H_z \Big|_{0,j,k}^n - 3H_z \Big|_{1,j,k}^{n-1} + H_z \Big|_{2,j,k}^{n-2} \quad (\text{Eq.45})$$

Em  $j = 0$ :

$$H_z \Big|_{i,-1,k}^n = 3H_z \Big|_{i,0,k}^n - 3H_z \Big|_{i,1,k}^{n-1} + H_z \Big|_{i,2,k}^{n-2} \quad (\text{Eq.46})$$

$$H_x \Big|_{i,-1,k}^n = 3H_x \Big|_{i,0,k}^n - 3H_x \Big|_{i,1,k}^{n-1} + H_x \Big|_{i,2,k}^{n-2} \quad (\text{Eq.47})$$

Em  $k = 0$ :

$$H_x \Big|_{i,j,-1}^n = 3H_x \Big|_{i,j,0}^n - 3H_x \Big|_{i,j,1}^{n-1} + H_x \Big|_{i,j,2}^{n-2} \quad (\text{Eq.48})$$

$$H_y \Big|_{i,j,-1}^n = 3H_y \Big|_{i,j,0}^n - 3H_y \Big|_{i,j,1}^{n-1} + H_y \Big|_{i,j,2}^{n-2} \quad (\text{Eq.49})$$

Em  $i = i_{\max}$ :

$$E_y \Big|_{i_{\max}+1,j,k}^n = 3E_y \Big|_{i_{\max},j,k}^n - 3E_y \Big|_{i_{\max}-1,j,k}^{n-1} + E_y \Big|_{i_{\max}-2,j,k}^{n-2} \quad (\text{Eq.50})$$

$$E_z \Big|_{i_{\max}+1,j,k}^n = 3E_z \Big|_{i_{\max},j,k}^n - 3E_z \Big|_{i_{\max}-1,j,k}^{n-1} + E_z \Big|_{i_{\max}-2,j,k}^{n-2} \quad (\text{Eq.51})$$

Em  $j = j_{\max}$ :

$$E_z \Big|_{i,j_{\max}+1,k}^n = 3E_z \Big|_{i,j_{\max},k}^n - 3E_z \Big|_{i,j_{\max}-1,k}^{n-1} + E_z \Big|_{i,j_{\max}-2,k}^{n-2} \quad (\text{Eq.52})$$

$$E_x \Big|_{i,j_{\max}+1,k}^n = 3E_x \Big|_{i,j_{\max},k}^n - 3E_x \Big|_{i,j_{\max}-1,k}^{n-1} + E_x \Big|_{i,j_{\max}-2,k}^{n-2} \quad (\text{Eq.53})$$

Em  $k = k_{\max}$ :

$$E_x \Big|_{i,j,k_{\max}+1}^n = 3E_x \Big|_{i,j,k_{\max}}^n - 3E_x \Big|_{i,j,k_{\max}-1}^{n-1} + E_x \Big|_{i,j,k_{\max}-2}^{n-2} \quad (\text{Eq.54})$$

$$E_y \Big|_{i,j,k_{\max}+1}^n = 3E_y \Big|_{i,j,k_{\max}}^n - 3E_y \Big|_{i,j,k_{\max}-1}^{n-1} + E_y \Big|_{i,j,k_{\max}-2}^{n-2} \quad (\text{Eq.55})$$

Contudo, para alcançar o referido erro de 0,5% é necessário trabalhar com número flutuante de precisão dupla. Essa hipótese conduziria a um aumento expressivo de memória RAM, necessária para armazenar os dados.

Por esse motivo, foi procurado um outro método de ABC que apresentasse uma melhor relação entre desempenho e consumo de recurso computacional.

#### **4.6.3 Meios Externos Absorventes (PML e UPML)**

A idéia é atribuir às bordas do *grid* características absorventes. Ou seja, ao invés de impedir que a onda atinja a borda, procura-se impedir que ela seja refletida.

Das várias técnicas apresentadas, tem recebido destaque a abordagem proposta por Berenger[6], denominada *Perfect Matched Layer (PML)* [1][2][3], devido aos resultados obtidos.

Nessa técnica, o *grid* é envolvido por um meio hipotético, de espessura entre 5 e 20 células, com propriedades elétricas e magnéticas adequadas para absorver a onda incidente. Dessa forma, a onda é atenuada antes de chegar ao final da camada de meio absorvente e, portanto, não sendo refletida para o *grid*.

Para que não exista dependência do ângulo de incidência da onda, Berenger propôs que cada componente de campo fosse dividida em duas subcomponentes, desdobrando as equações de Maxwell em 12 relações de diferenças finitas.

A técnica proposta por Berenger impõe uma alteração significativa no algoritmo de Yee, a fim de implementar o desdobramento das componentes de campo. Devido à sua complexidade surgiram outras propostas baseadas na mesma idéia da PML, entre as quais se destaca a *Uniaxial Perfect Matched Layer (UPML)*, analisada pela primeira vez por Sacks et al [7].

Embora, o meio e as subcomponentes propostos por Berenger sejam apenas abstrações numéricas que não tem correspondência no mundo real, percebeu-se que a dependência das propriedades com a direção é característica de meios anisotrópicos. Assim, a partir da aplicação de um modelo físico baseado em um meio anisotrópico e uniaxial, foi possível alcançar os mesmos resultados da técnica proposta por Berenger, sem alterar de forma significativa a formulação original de Yee.

A técnica foi desenvolvida inicialmente para um meio sem perdas (coeficientes de atenuação  $\sigma$  e  $\sigma^*$  nulos), o qual é então circundado pelo meio hipotético absorvente, cujas propriedades são convenientemente escolhidas para que a onda seja atenuada na UPML, sem haver reflexão. Além disso, os autores comprovam que as propriedades não reflexivas do meio hipotético que compõe a UPML são independentes do ângulo de incidência, polarização e frequência da onda incidente. A demonstração matemática da aplicação do conceito da UPML sobre o

FDTD está detalhada em Taflov [1], capítulo 7, a qual não é objeto desse trabalho, sendo pertinente ao contexto o equacionamento obtido, expresso abaixo no domínio da frequência:

$$\nabla \times \check{\mathbf{E}} = -j\omega \overset{=}{\mu} \check{\mathbf{H}} \quad (\text{Eq.56})$$

$$\nabla \times \check{\mathbf{H}} = j\omega \overset{=}{\varepsilon} \check{\mathbf{E}} \quad (\text{Eq.57})$$

onde:

$\check{\mathbf{E}}$  e  $\check{\mathbf{H}}$  – fasores associados aos vetores de campo

$\overset{=}{s}$  – tensor que aplica às características da *UPML*

$\varepsilon$  e  $\mu$  – permissividades elétrica e magnética do meio (*grid*)

Sendo os tensores definidos pelas seguintes expressões:

$$\overset{=}{s} = \begin{bmatrix} \frac{s_y s_z}{s_x} & 0 & 0 \\ 0 & \frac{s_x s_z}{s_y} & 0 \\ 0 & 0 & \frac{s_x s_y}{s_z} \end{bmatrix} \quad (\text{Eq.58})$$

$$s_x = k_x + \frac{\sigma_x}{j\omega\varepsilon} \quad (\text{Eq. 59})$$

$$s_y = k_y + \frac{\sigma_y}{j\omega\varepsilon} \quad (\text{Eq. 60})$$

$$s_z = k_z + \frac{\sigma_z}{j\omega\varepsilon} \quad (\text{Eq. 61})$$

onde:

$s_x$ ,  $s_y$  e  $s_z$  – definem as características da *UPML* nas direções  $x$ ,  $y$  e  $z$ , respectivamente

$k_x$ ,  $k_y$  e  $k_z$  – associados à permissividade ou permeabilidade da *UPML* relativa a permissividade ou permeabilidade do *grid* (normalmente de valor unitário)

$\sigma_x$ ,  $\sigma_y$  e  $\sigma_z$  – associadas à condutividade  $\sigma$  da *UPML*

Percebe-se que, ao multiplicar as equações Eq.59, Eq.60 e Eq.61 pela permissividade elétrica  $\epsilon$ , tem-se uma expressão semelhante ao que formalmente é definido como permissividade complexa do meio.

Da mesma forma, ao multiplicar as equações Eq.59, Eq.60 e Eq.61 pela permeabilidade magnética  $\mu$ , pode-se imaginar um conceito semelhante de permeabilidade complexa do meio.

Ainda, a partir da representação matricial dos vetores, as equações Eq.56 e Eq.57 podem ser reescritas da seguinte forma:

$$\begin{bmatrix} \frac{\partial \tilde{E}_z}{\partial y} - \frac{\partial \tilde{E}_y}{\partial z} \\ \frac{\partial \tilde{E}_x}{\partial z} - \frac{\partial \tilde{E}_z}{\partial x} \\ \frac{\partial \tilde{E}_y}{\partial x} - \frac{\partial \tilde{E}_x}{\partial y} \end{bmatrix} = -j\omega\mu \begin{bmatrix} \frac{s_y s_z}{s_x} & 0 & 0 \\ 0 & \frac{s_x s_z}{s_z} & 0 \\ 0 & 0 & \frac{s_y s_x}{s_z} \end{bmatrix} \begin{bmatrix} \tilde{H}_x \\ \tilde{H}_y \\ \tilde{H}_z \end{bmatrix} \quad (\text{Eq.62})$$

$$\begin{bmatrix} \frac{\partial \tilde{H}_z}{\partial y} - \frac{\partial \tilde{H}_y}{\partial z} \\ \frac{\partial \tilde{H}_x}{\partial z} - \frac{\partial \tilde{H}_z}{\partial x} \\ \frac{\partial \tilde{H}_y}{\partial x} - \frac{\partial \tilde{H}_x}{\partial y} \end{bmatrix} = j\omega\epsilon \begin{bmatrix} \frac{s_y s_z}{s_x} & 0 & 0 \\ 0 & \frac{s_x s_z}{s_z} & 0 \\ 0 & 0 & \frac{s_y s_x}{s_z} \end{bmatrix} \begin{bmatrix} \tilde{E}_x \\ \tilde{E}_y \\ \tilde{E}_z \end{bmatrix} \quad (\text{Eq.63})$$

Para fins desse trabalho, como na grande maioria dos casos estudados envolvendo o conceito da UPML, será definido que  $k_x = k_y = k_z = 1$ . Isso nos permite inferir uma série de considerações interessantes.

Nota-se que, se  $\overset{=}{s}$  for um tensor unitário ( $\sigma = 0$  em todas as direções), as equações Eq. 56 e Eq. 57 são reduzidas às equações de Maxwell (Eq. 5 e Eq. 6) no domínio frequência, para um meio sem perdas (coeficientes de atenuação  $\sigma$  e  $\sigma^*$  nulos).

De fato, o tensor  $\bar{s}$  é um artifício numérico para corrigir o problema de reflexão numérica, cujo valor pode ser estabelecido da forma mais conveniente para cada ponto do *grid*.

Assim sendo, no caso de um meio sem perdas, o tensor terá valor unitário para qualquer ponto interno ao *grid*, para que não sejam alteradas as propriedades do meio e a simulação através do algoritmo de Yee. Com efeito, a ação do tensor sobre as equações de campo existe apenas na camada de meio absorvente externa ao *grid*, isto é, somente na *UPML*.

Isso pode ser melhor compreendido através da tabela a seguir:

Tab. 6 – Valores dos coeficientes do tensor  $\bar{s}$  em função da posição no *grid*

Localização	$k_x$	$k_y$	$k_z$	$\sigma_x$	$\sigma_y$	$\sigma_z$
Interior do <i>grid</i> .	1	1	1	0	0	0
UPML – meio hipotético em $x < 0$ e $x > i_{\max}$	1	1	1	$\sigma_x$	0	0
UPML – meio hipotético em $y < 0$ e $y > y_{\max}$	1	1	1	0	$\sigma_y$	0
UPML – meio hipotético em $z < 0$ e $z > i_{\max}$	1	1	1	0	0	$\sigma_z$
UPML – intersecção dos meios hipotéticos em: <ul style="list-style-type: none"> <li>• <math>x &lt; 0</math> e <math>y &lt; 0</math></li> <li>• <math>x &lt; 0</math> e <math>y &gt; y_{\max}</math></li> <li>• <math>x &gt; x_{\max}</math> e <math>y &lt; 0</math></li> <li>• <math>x &gt; x_{\max}</math> e <math>y &gt; y_{\max}</math></li> </ul>	1	1	1	$\sigma_x$	$\sigma_y$	0
UPML – intersecção dos meios hipotéticos em: <ul style="list-style-type: none"> <li>• <math>x &lt; 0</math> e <math>z &lt; 0</math></li> <li>• <math>x &lt; 0</math> e <math>z &gt; z_{\max}</math></li> <li>• <math>x &gt; x_{\max}</math> e <math>z &lt; 0</math></li> <li>• <math>x &gt; x_{\max}</math> e <math>z &gt; z_{\max}</math></li> </ul>	1	1	1	$\sigma_x$	0	$\sigma_z$
UPML – intersecção dos meios hipotéticos em: <ul style="list-style-type: none"> <li>• <math>z &lt; 0</math> e <math>y &lt; 0</math></li> <li>• <math>z &lt; 0</math> e <math>y &gt; y_{\max}</math></li> <li>• <math>z &gt; z_{\max}</math> e <math>y &lt; 0</math></li> <li>• <math>z &gt; z_{\max}</math> e <math>y &gt; y_{\max}</math></li> </ul>	1	1	1	0	$\sigma_y$	$\sigma_z$
UPML – intersecção dos meios hipotéticos nos cantos	1	1	1	$\sigma_x$	$\sigma_y$	$\sigma_z$

Porém, devido a discretização do meio em células de dimensões finitas, impõe-se que a transição do meio no qual é composto o *grid* para o meio hipotético no qual é formado a *UPML* não ocorra de forma abrupta. Em termos mais claros, os coeficientes  $\sigma$  não podem variar de zero (meio sem perdas) para o valor final imediatamente na primeira célula da *UPML*.

Conforme proposto por Berenger [6], para resolver tal situação, deve-se fazer  $\sigma$  variar conforme se avança através da *UPML*. Segundo Taflov [1], com base na coletânea de trabalhos e experimentos de diversos autores, o perfil de variação polinomial é um dos mais adotados, o qual é regido pela expressão que segue:

$$\sigma_w(w) = \sigma_{\max} \left( \frac{w}{d} \right)^m \quad (\text{Eq.64})$$

onde:

$w$  – distância percorrida no interior da *UPML*

$d = N_{upml} \cdot \Delta$  – espessura da *UPML*

$N_{upml}$  – número de células da *UPML*

$\sigma_w$  – refere-se à condutividade da *UPML* em uma das direções

$m$  – expoente, com valor usual compreendido entre 3 e 4

Para completar a solução, é apresentada a relação entre a performance da *UPML* e o valor  $\sigma_{\max}$ , estabelecida pela expressão:

$$\sigma_{\max} = - \frac{(m+1) \cdot \ln R}{2\eta d} \quad (\text{Eq.65})$$

onde:

$$\eta = \sqrt{\mu/\varepsilon}$$

$$d = N_{upml} \cdot \Delta$$

$R$  – coeficiente de reflexão numérico teórico

Apesar de ser desejado  $R = 0$ , experimentos numéricos demonstraram que valores muito elevados de  $\sigma$  implicam no aparecimento de erro numérico associado a discretização (a variação de  $\sigma$  não é suficientemente gradual), o que torna a *UPML* ineficiente. Assim, a partir da realização de vários experimentos numéricos, foram estabelecidos valores adequados de  $R$ , para os quais a Eq.65 é válida.

Tab. 7 – Coeficiente de reflexão numérico em função do número de células da *UPML*

Número de células da <i>UPML</i>	$R$
5	$e^{-8} (\approx 3.10^{-4})$
10	$e^{-16} (\approx 1.10^{-7})$

#### 4.6.3.1 UMPL e o algoritmo de Yee (meio sem perdas)

Para chegar a formulação compatível com o algoritmo de Yee é necessário passar as equações Eq.62 e Eq.63 para o domínio do tempo, o que é realizado através da transformada inversa de Fourier aplicada as equações

Porém, para facilitar na solução da transformada inversa, são definidas as seguintes variáveis auxiliares:

$$\tilde{E}_x = \frac{s_x}{s_z} \tilde{D}_x \quad (\text{Eq.66})$$

$$\tilde{E}_y = \frac{s_y}{s_x} \tilde{D}_y \quad (\text{Eq.67})$$

$$\tilde{E}_z = \frac{s_z}{s_y} \tilde{D}_z \quad (\text{Eq.68})$$

$$\tilde{H}_x = \frac{s_x}{s_z} \tilde{B}_x \quad (\text{Eq.69})$$

$$\tilde{H}_y = \frac{s_y}{s_x} \tilde{B}_y \quad (\text{Eq.70})$$

$$\tilde{H}_z = \frac{s_z}{s_y} \tilde{B}_z \quad (\text{Eq.71})$$

A substituição das variáveis auxiliares nas equações Eq.62 e Eq.63 resulta no equacionamento que segue:

$$\begin{bmatrix} \frac{\partial \tilde{E}_z}{\partial y} - \frac{\partial \tilde{E}_y}{\partial z} \\ \frac{\partial \tilde{E}_x}{\partial z} - \frac{\partial \tilde{E}_z}{\partial x} \\ \frac{\partial \tilde{E}_y}{\partial x} - \frac{\partial \tilde{E}_x}{\partial y} \end{bmatrix} = -j\omega\mu \begin{bmatrix} s_y & 0 & 0 \\ 0 & s_z & 0 \\ 0 & 0 & s_x \end{bmatrix} \begin{bmatrix} \tilde{B}_x \\ \tilde{B}_y \\ \tilde{B}_z \end{bmatrix} \quad (\text{Eq.72})$$

$$\begin{bmatrix} \frac{\partial \tilde{H}_z}{\partial y} - \frac{\partial \tilde{H}_y}{\partial z} \\ \frac{\partial \tilde{H}_x}{\partial z} - \frac{\partial \tilde{H}_z}{\partial x} \\ \frac{\partial \tilde{H}_y}{\partial x} - \frac{\partial \tilde{H}_x}{\partial y} \end{bmatrix} = j\omega\varepsilon \begin{bmatrix} s_y & 0 & 0 \\ 0 & s_z & 0 \\ 0 & 0 & s_x \end{bmatrix} \begin{bmatrix} \tilde{D}_x \\ \tilde{D}_y \\ \tilde{D}_z \end{bmatrix} \quad (\text{Eq.73})$$

Ainda, ao substituir os componentes dos tensores pelas expressões que os definem, tem-se:

$$\begin{bmatrix} \frac{\partial \tilde{E}_z}{\partial y} - \frac{\partial \tilde{E}_y}{\partial z} \\ \frac{\partial \tilde{E}_x}{\partial z} - \frac{\partial \tilde{E}_z}{\partial x} \\ \frac{\partial \tilde{E}_y}{\partial x} - \frac{\partial \tilde{E}_x}{\partial y} \end{bmatrix} = -j\omega\mu \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{B}_x \\ \tilde{B}_y \\ \tilde{B}_z \end{bmatrix} - \begin{bmatrix} \sigma_y & 0 & 0 \\ 0 & \sigma_z & 0 \\ 0 & 0 & \sigma_x \end{bmatrix} \begin{bmatrix} \tilde{B}_x \\ \tilde{B}_y \\ \tilde{B}_z \end{bmatrix} \quad (\text{Eq.73})$$

$$\begin{bmatrix} \frac{\partial \tilde{H}_z}{\partial y} - \frac{\partial \tilde{H}_y}{\partial z} \\ \frac{\partial \tilde{H}_x}{\partial z} - \frac{\partial \tilde{H}_z}{\partial x} \\ \frac{\partial \tilde{H}_y}{\partial x} - \frac{\partial \tilde{H}_x}{\partial y} \end{bmatrix} = j\omega\varepsilon \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{D}_x \\ \tilde{D}_y \\ \tilde{D}_z \end{bmatrix} + \begin{bmatrix} \sigma_y & 0 & 0 \\ 0 & \sigma_z & 0 \\ 0 & 0 & \sigma_x \end{bmatrix} \begin{bmatrix} \tilde{D}_x \\ \tilde{D}_y \\ \tilde{D}_z \end{bmatrix} \quad (\text{Eq.74})$$

E finalmente, aplicando a transformada inversa de Fourier, tem-se:

$$\frac{\partial}{\partial t} \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} = -\frac{1}{\mu} \begin{bmatrix} \frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z} \\ \frac{\partial E_x}{\partial z} - \frac{\partial E_z}{\partial x} \\ \frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y} \end{bmatrix} - \frac{1}{\mu} \begin{bmatrix} \sigma_y & 0 & 0 \\ 0 & \sigma_z & 0 \\ 0 & 0 & \sigma_x \end{bmatrix} \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} \quad (\text{Eq.75})$$

$$\frac{\partial}{\partial t} \begin{bmatrix} D_x \\ D_y \\ D_z \end{bmatrix} = \frac{1}{\varepsilon} \begin{bmatrix} \frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} \\ \frac{\partial H_x}{\partial z} - \frac{\partial H_z}{\partial x} \\ \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} \end{bmatrix} - \frac{1}{\varepsilon} \begin{bmatrix} \sigma_y & 0 & 0 \\ 0 & \sigma_z & 0 \\ 0 & 0 & \sigma_x \end{bmatrix} \begin{bmatrix} D_x \\ D_y \\ D_z \end{bmatrix} \quad (\text{Eq.76})$$

Ao comparar com as equações Eq.5a e Eq.6a, percebe-se o formato idêntico, que resultará em equacionamento análogo do algoritmo de Yee ao serem aplicadas as relações de diferenças finitas, porém envolvendo agora as variáveis auxiliares.

Para fins de exemplificação, pode-se analisar o equacionamento para a determinação da componente de campo  $E_x$ . A partir da equação Eq.75, de forma análoga a Eq.33, pode-se demonstrar que:

$$D_x \Big|_{i,j,k}^{n+1} = Ca \Big|_{i,j,k}^y D_x \Big|_{i,j,k}^n + Cb \Big|_{i,j,k}^y \left( H_z \Big|_{i,j,k}^n - H_z \Big|_{i,j-1,k}^n + H_y \Big|_{i,j,k-1}^n - H_y \Big|_{i,j,k}^n \right) \quad (\text{Eq.77})$$

onde:  $Ca \Big|_{i,j,k}^y$  e  $Cb \Big|_{i,j,k}^y$  são as mesmas expressões de  $Ca_{i,j,k}$  e  $Cb_{i,j,k}$  (Eq.25 e Eq.26), calculadas utilizando  $\sigma_y$ .

Então, para calcular o valor da componente  $E_x$ , faz-se necessário o uso da equação de definição da variável auxiliar (no caso, Eq.66), de onde tem-se:

$$\left( 1 + \frac{\sigma_x}{j\omega\varepsilon} \right) \check{D}_x = \left( 1 + \frac{\sigma_z}{j\omega\varepsilon} \right) \check{E}_x \quad (\text{Eq.78})$$

E no domínio do tempo, tem-se:

$$\varepsilon \frac{\partial D_x}{\partial t} + \sigma_x D_x = \varepsilon \frac{\partial E_x}{\partial t} + \sigma_z E_x \quad (\text{Eq.79})$$

Ao aplicar as relações de diferenças finitas, utilizando a notação apresentada no equacionamento básico, obtém-se a expressão para determinação de  $E_x$ :

$$E_x \Big|_{i,j,k}^{n+1} = Ka \Big|_{i,j,k}^z E_x \Big|_{i,j,k}^n + Kb \Big|_{i,j,k}^{x,z} D_x \Big|_{i,j,k}^{n+1} - Kc \Big|_{i,j,k}^{x,z} D_x \Big|_{i,j,k}^n \quad (\text{Eq.80})$$

onde:

$$Ka \Big|_{i,j,k}^z = \frac{1 - \frac{\sigma_z \Delta t}{2\varepsilon}}{1 + \frac{\sigma_z \Delta t}{2\varepsilon}} \quad (\text{Eq.81})$$

$$Kb \Big|_{i,j,k}^{x,z} = \frac{1 + \frac{\sigma_x \Delta t}{2\varepsilon}}{1 + \frac{\sigma_z \Delta t}{2\varepsilon}} \quad (\text{Eq.82})$$

$$Kc|_{i,j,k}^{x,z} = \frac{1 - \frac{\sigma_x \Delta t}{2\varepsilon}}{1 + \frac{\sigma_z \Delta t}{2\varepsilon}} \quad (\text{Eq.83})$$

Portanto, fica evidente que o algoritmo de Yee é mantido, com a exigência de um passo adicional para determinação da variável auxiliar. Ressalta-se que todas as considerações estabelecidas para avaliação de  $\sigma_w$  são mantidas.

Contudo o equacionamento apresentado é para um *grid* composto por um meio sem perdas, que naturalmente limita muito a aplicação do algoritmo de cálculo. A seguir é explorado como estender o conceito para meio com perdas.

#### 4.6.3.2 UMPL e o algoritmo de Yee (meio com perdas)

Para um meio com perdas, obviamente é necessário considerar  $\sigma \neq 0$  e  $\sigma^* \neq 0$  ao longo de todo o *grid*. Com efeito, as Eq.62 e Eq.63 tomam a seguinte forma:

$$\begin{bmatrix} \frac{\partial \tilde{E}_z}{\partial y} - \frac{\partial \tilde{E}_y}{\partial z} \\ \frac{\partial \tilde{E}_x}{\partial z} - \frac{\partial \tilde{E}_z}{\partial x} \\ \frac{\partial \tilde{E}_y}{\partial x} - \frac{\partial \tilde{E}_x}{\partial y} \end{bmatrix} = -(j\omega\mu + \sigma^*) \begin{bmatrix} \frac{s_y s_z}{s_x} & 0 & 0 \\ 0 & \frac{s_x s_z}{s_z} & 0 \\ 0 & 0 & \frac{s_y s_x}{s_z} \end{bmatrix} \begin{bmatrix} \tilde{H}_x \\ \tilde{H}_y \\ \tilde{H}_z \end{bmatrix} \quad (\text{Eq.84})$$

$$\begin{bmatrix} \frac{\partial \tilde{H}_z}{\partial y} - \frac{\partial \tilde{H}_y}{\partial z} \\ \frac{\partial \tilde{H}_x}{\partial z} - \frac{\partial \tilde{H}_z}{\partial x} \\ \frac{\partial \tilde{H}_y}{\partial x} - \frac{\partial \tilde{H}_x}{\partial y} \end{bmatrix} = (j\omega\varepsilon + \sigma) \begin{bmatrix} \frac{s_y s_z}{s_x} & 0 & 0 \\ 0 & \frac{s_x s_z}{s_z} & 0 \\ 0 & 0 & \frac{s_y s_x}{s_z} \end{bmatrix} \begin{bmatrix} \tilde{E}_x \\ \tilde{E}_y \\ \tilde{E}_z \end{bmatrix} \quad (\text{Eq.85})$$

Nesse caso, para realizar o equacionamento adaptado ao algoritmo de Yee são definidas as seguintes variáveis auxiliares:

$$\tilde{E}_x = \frac{s_x}{s_y s_z} \tilde{F}_x \quad (\text{Eq.86a}) \quad \text{e} \quad \tilde{F}_x = s_y \tilde{G}_x \quad (\text{Eq.86b})$$

$$\tilde{E}_y = \frac{s_y}{s_x s_z} \tilde{F}_y \quad (\text{Eq.87a}) \quad \text{e} \quad \tilde{F}_y = s_z \tilde{G}_y \quad (\text{Eq.87b})$$

$$\tilde{E}_z = \frac{s_z}{s_x s_y} \tilde{F}_z \quad (\text{Eq.88a}) \quad \text{e} \quad \tilde{F}_z = s_x \tilde{G}_z \quad (\text{Eq.88b})$$

$$\tilde{H}_x = \frac{s_x}{s_y s_z} \tilde{P}_x \quad (\text{Eq.89a}) \quad \text{e} \quad \tilde{P}_x = s_y \tilde{Q}_x \quad (\text{Eq.89b})$$

$$\tilde{H}_y = \frac{s_y}{s_x s_z} \tilde{P}_y \quad (\text{Eq.90a}) \quad \text{e} \quad \tilde{P}_y = s_z \tilde{Q}_y \quad (\text{Eq.90b})$$

$$\tilde{H}_z = \frac{s_z}{s_x s_y} \tilde{P}_z \quad (\text{Eq.91a}) \quad \text{e} \quad \tilde{P}_z = s_x \tilde{Q}_z \quad (\text{Eq.91b})$$

A substituição das variáveis auxiliares  $F$  e  $P$  nas equações Eq.84 e Eq.85 resulta no equacionamento que segue:

$$\begin{bmatrix} \frac{\partial \tilde{E}_z}{\partial y} - \frac{\partial \tilde{E}_y}{\partial z} \\ \frac{\partial \tilde{E}_x}{\partial z} - \frac{\partial \tilde{E}_z}{\partial x} \\ \frac{\partial \tilde{E}_y}{\partial x} - \frac{\partial \tilde{E}_x}{\partial y} \end{bmatrix} = -(j\omega\mu + \sigma^*) \begin{bmatrix} \tilde{P}_x \\ \tilde{P}_y \\ \tilde{P}_z \end{bmatrix} \quad (\text{Eq.92})$$

$$\begin{bmatrix} \frac{\partial \tilde{H}_z}{\partial y} - \frac{\partial \tilde{H}_y}{\partial z} \\ \frac{\partial \tilde{H}_x}{\partial z} - \frac{\partial \tilde{H}_z}{\partial x} \\ \frac{\partial \tilde{H}_y}{\partial x} - \frac{\partial \tilde{H}_x}{\partial y} \end{bmatrix} = (j\omega\varepsilon + \sigma) \begin{bmatrix} \tilde{F}_x \\ \tilde{F}_y \\ \tilde{F}_z \end{bmatrix} \quad (\text{Eq.93})$$

Ressalta-se aqui que  $\sigma$  e  $\sigma^*$  são as propriedades do meio do qual é composto o *grid*. Nesse caso, faz-se necessário estender as propriedades dos materiais nas bordas do *grid* para as células da *UPML*, para manutenção da continuidade ao longo do espaço de simulação (*grid* + *UPML*).

Por outro lado, as propriedades  $\sigma_w$  da UPML são tratadas exclusivamente nas equações resultantes das expressões de definição das variáveis auxiliares.

Então, aplicando a transformada inversa de Fourier às equações Eq.92 e Eq.93, tem-se:

$$\frac{\partial}{\partial t} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = -\frac{1}{\mu} \begin{bmatrix} \frac{\partial E_z}{\partial y} - \frac{\partial E_y}{\partial z} \\ \frac{\partial E_x}{\partial z} - \frac{\partial E_z}{\partial x} \\ \frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y} \end{bmatrix} - \frac{\sigma^*}{\mu} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \quad (\text{Eq.94})$$

$$\frac{\partial}{\partial t} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} = \frac{1}{\varepsilon} \begin{bmatrix} \frac{\partial H_z}{\partial y} - \frac{\partial H_y}{\partial z} \\ \frac{\partial H_x}{\partial z} - \frac{\partial H_z}{\partial x} \\ \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} \end{bmatrix} - \frac{\sigma}{\varepsilon} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \quad (\text{Eq.95})$$

Novamente, ao comparar com as equações Eq.5a e Eq.6a, percebe-se o formato idêntico, que resultará em equacionamento análogo do algoritmo de Yee ao serem aplicadas as relações de diferenças finitas, porém envolvendo agora as variáveis auxiliares  $F$  e  $P$ .

Outra vez, para fins de exemplificação, pode-se analisar o equacionamento para a determinação da componente de campo  $E_x$ . A partir da equação Eq.95, de forma análoga à Eq.33, pode-se demonstrar que:

$$F_x|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot F_x|_{i,j,k}^n + Cb_{i,j,k} \left( H_z|_{i,j,k}^n - H_z|_{i,j-1,k}^n + H_y|_{i,j,k-1}^n - H_y|_{i,j,k}^n \right) \quad (\text{Eq.96})$$

onde:  $Ca_{i,j,k}$  e  $Cb_{i,j,k}$  são calculadas pelas equação Eq.25 e Eq.26.

Fazendo o uso da equação de definição das variáveis auxiliares (no caso, Eq.86b), tem-se:

$$F_x = \left( 1 + \frac{\sigma_y}{j\omega\varepsilon} \right) G_x \quad (\text{Eq.97})$$

E no domínio do tempo, tem-se:

$$\frac{\partial F_x}{\partial t} = \frac{\partial G_x}{\partial t} + \frac{\sigma_y}{\varepsilon} G_x \quad (\text{Eq.98})$$

Ao aplicar as relações de diferenças finitas, utilizando a notação apresentada no equacionamento básico, obtém-se a expressão:

$$G_x|_{i,j,k}^{n+1} = Kd|_{i,j,k}^y G_x|_{i,j,k}^n + Ke|_{i,j,k}^y \left( F_x|_{i,j,k}^{n+1} - F_x|_{i,j,k}^n \right) \quad (\text{Eq.99})$$

onde:

$$Kd|_{i,j,k}^y = \frac{1 - \frac{\sigma_y \Delta t}{2\varepsilon}}{1 + \frac{\sigma_y \Delta t}{2\varepsilon}} \quad (\text{Eq.100})$$

$$Ke|_{i,j,k}^y = \frac{1}{1 + \frac{\sigma_y \Delta t}{2\varepsilon}} \quad (\text{Eq.101})$$

Ainda, combinando as equações Eq.86a e Eq.86b, tem-se:

$$\tilde{E}_x = \frac{s_x}{s_z} \tilde{G}_x \quad (\text{Eq.86c})$$

Sendo essa equação análoga a Eq.66, pode-se concluir que:

$$E_x|_{i,j,k}^{n+1} = Ka|_{i,j,k}^z E_x|_{i,j,k}^n + Kb|_{i,j,k}^{x,z} G_x|_{i,j,k}^{n+1} - Kc|_{i,j,k}^{x,z} G_x|_{i,j,k}^n \quad (\text{Eq.102})$$

Portanto, fica evidente que o algoritmo de Yee é mantido, com a exigência de dois passos adicionais para determinação das variáveis auxiliares. Ressalta-se que todas as considerações estabelecidas para avaliação de  $\sigma_w$  são mantidas.

São válidas também todas as suposições sobre os incrementos  $\Delta$  e  $\Delta t$ , que fundamentaram a obtenção das equações Eq.33 a Eq.38.

## 5. Proposta de Trabalho

Desenvolver programa que implemente a simulação de sistemas pelo *FDTD*, em três dimensões, que possa ser utilizado em computadores de uso geral.

Implementar o algoritmo de Yee original (sem técnicas para absorção da onda na bordas do *grid*) e o algoritmo de Yee modificado com a técnica da *UPML*, comparando os resultados obtidos, considerando:

- recursos computacionais exigidos
- tempo de processamento
- resultados de simulação obtidos, com foco na redução dos problemas de reflexão.

Desenvolver técnicas e algoritmos que permitam a otimização do processo de cálculo iterativo e a estruturação do código proposto para viabilizar sua utilização futura na simulação de sistemas muito complexos, considerando a hipótese do emprego de técnicas de computação paralela.

## 6. Metodologia

### 6.1 Hipóteses ao equacionamento

Cada célula é formada por um único material, o qual é homogêneo. Os materiais e suas propriedades encontram-se previamente cadastrados em uma lista indexada pelo argumento  $m$ , único para cada material da lista.

Dessa forma, o espaço (o *grid*) é caracterizado pela determinação de  $\Delta$ , do número de células em cada dimensão ( $N_x$ ,  $N_y$  e  $N_z$ ) e do material presente em cada célula. Essas informações são armazenadas em disco, sendo a última (material em cada célula) organizada em  $N_x$  matrizes  $N_y \times N_z$  elementos, contendo cada um o valor de  $m$  correspondente àquela célula.

Conhecido o material existente em cada célula, pode-se facilmente calcular o valor dos coeficientes  $Ca$ ,  $Cb$ ,  $Da$  e  $Db$ , através das equações (Eq.25) a (Eq.28).

### 6.2 Algoritmo básico e recursos computacionais

Com o equacionamento básico definido, pode-se estabelecer o algoritmo básico de cálculo e, a partir desse, avaliar os recursos computacionais mínimos para executá-lo.

Assim sendo, conforme equacionamento apresentado, algoritmo básico de cálculo, desconsiderando ainda o problema de reflexão nas bordas e a existência das fontes de campo, tem a seguinte forma:

1. Pré-processamento:
  - 1.1. Definir as propriedades eletromagnéticas dos materiais envolvidos
  - 1.2. Definir o tamanho do *grid* e o material presente em cada célula do *grid*
  - 1.3. Definir o instante de tempo final, quando será interrompido o processo de cálculo
2. Processamento inicial:
  - 2.1. Definir os valores de  $\Delta$  e  $\Delta t$  (Eq. 29 e 30)
  - 2.2. Estabelecer os valores iniciais de campo elétrico e magnético em cada célula

3. Para cada célula  $i,j,k$  do *grid*, a partir da primeira (processamento do campo elétrico):
  - 3.1. Ler o índice  $m$  do material
  - 3.2. Ler as propriedades eletromagnéticas ao material de índice  $m$
  - 3.3. Calcular os valores dos coeficientes  $Ca_{i,j,k}$  e  $Cb_{i,j,k}$  (Eq. 25 e 26)
  - 3.4. Calcular o valor das fontes de campo elétrico presentes na célula, no instante  $n$
  - 3.5. Ler os valores das componentes de campo magnético, avaliadas para o instante  $n$
  - 3.6. Calcular os valores das componentes de campo elétrico para o próximo instante de tempo  $n+1$  (Eq. 33 a 35). Impor os valores das fontes de campo elétrico, quando aplicável
  - 3.7. Armazenar os valores das componentes de campo elétrico calculados
  
4. Para cada célula  $i,j,k$  do *grid*, a partir da primeira (processamento do campo magnético):
  - 4.1. Ler o índice  $m$  do material
  - 4.2. Ler as propriedades eletromagnéticas ao material de índice  $m$
  - 4.3. Calcular os valores dos coeficientes  $Da_{i,j,k}$  e  $Db_{i,j,k}$  (Eq. 27 e 28)
  - 4.4. Calcular o valor das fontes de campo magnético presentes na célula, no instante  $n$
  - 4.5. Ler os valores das componentes de campo elétrico, avaliadas para o instante  $n+1$
  - 4.6. Calcular os valores das componentes de campo magnético para o próximo instante de tempo  $n+1$  (Eq. 36 a 38). Impor os valores das fontes de campo magnético, quando aplicável
  - 4.7. Armazenar os valores das componentes de campo magnético calculados
  
5. Incrementar tempo. Caso não tenha alcançado tempo final, voltar para (3).
6. Apresentar resultados.

A etapa de pré-processamento efetivamente consiste em definir o problema de engenharia e em realizar a coleta dos dados necessários para cálculo.

Os dados coletados são armazenados em arquivos ou então utilizados como parâmetros de entrada do algoritmo de cálculo propriamente dito, realizado a partir da etapa 2.

Considerando o objetivo do trabalho (uma ferramenta computacional para um único computador de uso geral), é de interesse que pelo menos os seguintes aspectos básicos sejam avaliados.

- a) Há memória RAM suficiente para executar o programa?
- b) A memória ocupada pelos arquivos em disco é compatível com as atuais mídias de armazenamento?

- c) O tempo de processamento é adequado, ou seja, o programa não será excessivamente lento?

Como ponto de partida para responder tais questionamentos, algumas premissas precisam ser estabelecidas para caracterizar o ambiente de execução do programa:

- computador alvo: Pentium 4 1,5 GHz, RAM 512 MB, HD 40 GB;
- linguagem de programação: ANSI C;
- compilador: optou-se por versões de software livre. Foram utilizados o DJGPP[13] e o Mingw[14], que acompanha a IDE Dev-C++[15];
- sistema operacional: Windows 2000 e XP. Embora não seja um software livre, está presente na maioria dos PCs de uso geral. Contudo, não existe empecilho em utilizar o Linux ou outro sistema operacional 32bits;
- o programa será uma aplicação do tipo console;

Estabelecido o ambiente de execução, é necessário avaliar a exigência computacional para executar o programa.

Para tanto, pode-se iniciar pela avaliação da quantidade mínima de variáveis associada a cada célula, a qual está apresentada na tabela abaixo, tomando como referência as relações apresentadas no equacionamento básico.

Tab. 8 – Quantidade variáveis associadas a cada célula

Descrição	tipo inteiro	tipo ponto flutuante
índice do material	01	-
propriedades eletromagnéticas do material	-	04
componentes de campo elétrico	-	03
componentes de campo magnético	-	03

Como os recursos de memória RAM e armazenamento são escassos, existe a tendência de se usar os tipos de menor tamanho possível. Assim, para o índice de material, considerou-se aceitável a existência de 256 tipos de materiais e adotou-se variáveis do tipo *char* (1 byte) para representar o índice *m*. Da mesma forma, o tipo *float* (4 bytes – ponto flutuante de precisão simples – 7 algarismos significativos) é a

escolha indicada. Logo, o espaço mínimo de memória ocupado por cada célula é de 41 bytes.

Assim, se admitido que o *grid* seja formado por 1000 matrizes de 1000 x 1000 células ( $10^9$  células), são necessários 41.000.000.000 bytes (aprox. 39.100 MB) de memória RAM, o que é nosso primeiro obstáculo, pois esse valor é 75 vezes maior que a memória RAM disponível.

Não sendo possível carregar tudo na RAM, tem-se que adotar uma estratégia para trabalhar com os valores armazenados em disco. Se o algoritmo básico de cálculo for seguido, exatamente como escrito, a memória RAM necessária seria muito pequena. Nessa hipótese de processamento, para cada componente de campo a ser calculada, seriam realizadas apenas a leitura das respectivas variáveis efetivamente necessárias naquele instante.

Em contra-partida, o número de operações de acesso a disco seria enorme, pois para cada célula e em cada interação de tempo seriam realizadas pelo menos as seguintes operações:

- leitura do índice de material: 02
- leitura das propriedades do material: 02
- leitura das componentes necessárias para cálculo: 18
- escrita dos valores calculados: 06

Por hipótese, as variáveis necessárias não estariam em um mesmo setor ou em setores contíguos do disco. Isso implica em que cada operação de leitura seria correspondente a uma operação de busca no disco, o que efetivamente é a parte mais demorada do processo de acesso a disco.

Ou seja, considerando o *grid* proposto, seriam necessárias  $28 \cdot 10^9$  operações de acesso a disco para cada interação de tempo. Se o tempo da operação de acesso fosse  $0,1 \text{ ms}^1$  então, seriam necessárias  $2,8 \cdot 10^6$  s (por interação), ou seja, algo em torno de 778 horas por interação de tempo, o que obviamente não serve.

---

<sup>1</sup> O valor adotado é muito menor que o tempo de busca médio nos dispositivos atuais. Nas referências [16] e [17] pode-se ter informações detalhadas sobre o funcionamento dos discos rígidos

Mesmo que a taxa de transferência dos dispositivos atuais ultrapasse os 100 MB/s, para o acesso a dados armazenados de forma contígua em disco, diversas operações de busca para a leitura de pequenas quantidades de dados torna o sistema inviável.

O problema foi resolvido pelo desenvolvimento de uma nova técnica de programação, a qual foi denominada “organização de variáveis por plano espacial”.

### **6.2.1 Organização de variáveis por plano espacial**

A técnica foi assim denominada por orientar a implementação do algoritmo de cálculo em função da distribuição das variáveis no espaço físico.

É evidente que a técnica pode ter sua aplicação generalizada para qualquer tipo de sistema cuja formulação esteja baseada na discretização do espaço tridimensional em elementos finitos e o cálculo sucessivo das propriedades de interesse ao longo do espaço-tempo. Contudo, a referida técnica será apresentada sob o enfoque do FDTD.

A partir da observação das equações, percebe-se que as variáveis envolvidas no cálculo de cada componente estão em células contíguas. E mais, ao ser percorrido um plano do *grid* (uma das matrizes de  $N_y \times N_z$  elementos), a fim de inspecionar as variáveis necessárias para cálculo de cada componente pertencente ao plano, percebe-se que as variáveis necessárias ou estão no mesmo plano, ou em um plano contíguo ao inspecionado.

Então, a estratégia adota foi organizar as variáveis de processo por plano do *grid*, de tal sorte que essas fossem armazenadas áreas contíguas de memória e em setores contíguo do disco. Assim, em poucas operações de acesso a disco, todas as variáveis necessárias para o cálculo das componentes de campo associadas a um determinado plano ou conjunto de planos são transferidas para a RAM e, após

realizados os cálculos, os novos valores são atualizados em uma operação de escrita.

Dessa forma, a técnica de organização de variáveis por plano espacial tem como objetivo disciplinar a ocupação de RAM e o acesso a disco. Pois, a hipótese de não controlar o tamanho de RAM utilizada e permitir ao sistema operacional utilizar a *cache* de disco para aumentar a capacidade de memória resultou em um código mais lento, devido ao acesso freqüente ao disco durante o cálculo iterativo.

Observa-se, porém, que o objetivo principal deve ser a busca por utilizar o maior espaço de RAM disponível, uma vez que o acesso a disco é extremamente lento em comparação ao acesso a RAM.

Considerando sistemas muito complexos, pode-se ainda considerar hipótese de utilizar técnicas de computação paralela, trazidas à realidade na década de 90, na forma de um pacote de software denominado *Parallel Virtual Machine (PVM)*, o qual permite agrupar diversos computadores de uma rede em uma estrutura denominada *cluster*.

De forma bastante simplória, o resultado do *PVM* é uma máquina virtual com diversos processadores trabalhando simultaneamente, cuja memória RAM total é a soma das memórias disponibilizadas pelas máquinas do *cluster*. Assim, seria possível ter uma memória tão grande quanto o desejável, sem a necessidade de acesso à disco.

Com esse enfoque, a técnica organização de variáveis por plano espacial traz o benefício de pré-organizar a realização do processamento paralelo. Por hipótese, o *grid* pode ser subdividido em bloco de N planos contíguos (*subgrids*), sendo cada *subgrid* processado em um dos computadores do *cluster*, de forma simulatânea, cabendo ao software *PVM* o gerenciamento dos dados dos planos de interface entre *subgrid*. Assim, embora não faça proposta de trabalho estudar e implementar técnicas de processamento paralelo, percebe-se a facilidade para implementação, a partir da aplicação da técnica organização de variáveis por plano espacial.

O primeiro passo da técnica proposta consiste em criar estruturas de dados adequadas. No caso do FDTD são definidas as seguintes estruturas, apresentadas na sintaxe da linguagem de programação adotada:

a) campos elétricos ou magnéticos:

```
typedef struct {  
    float x, y, z;  
}FIELD;
```

b) características do *grid*:

```
typedef struct {  
    char mat, fonte;  
}GRID;
```

Como o número de materiais e tipos de fontes em um plano é muito menor que o número de células, optou-se por definir as propriedades dos materiais e os parâmetros de cálculo das fontes existentes em estruturas auxiliares, externas a estrutura que define as características dos *grid*. São elas:

a) propriedades dos materiais:

```
typedef struct {  
    float Ca, Cb, Da, Db;  
}COEFIC;
```

b) fontes:

```
typedef struct {  
    unsigned int x, y, z, func;  
    char campo;  
    float par1,par2,par3;  
}SOURCE;
```

Optou-se por armazenar os valores pré-calculados dos coeficientes como integrantes das propriedades dos materiais, já que por hipótese não são alterados durante as interações de tempo. Isso tem por objetivo acelerar a execução do código, pois são eliminadas diversas operações matemáticas que seriam realiza

durante processo iterativo. Assim, o cálculo dos coeficientes resume-se ao acesso de memória, indexado pelo material presente em cada célula do *grid*.

A partir das estruturas definidas, são alocadas áreas de memória para atender as variáveis necessárias ao processamento. No caso do FDTD, tem-se:

- a)  $E [N][N_y][N_z]$  – N matrizes de  $N_y \times N_z$  elementos do tipo FIELD, sendo:
- $E [i][j][k].x$  – corresponde a  $E_x(j,k)$  do plano “i”
  - $E [i][j][k].y$  – corresponde a  $E_y(j,k)$  do plano “i”
  - $E [i][j][k].z$  – corresponde a  $E_z(j,k)$  do plano “i”

Os valores futuros de campo elétrico em cada posição são armazenados na mesma variável que contém os valores presentes, pela atualização de seu valor. Isso é possível pela forma como as operações de cálculo e variáveis necessárias são processadas no algoritmo.

O valor de N pode ou não ser igual à  $N_x$ , dependendo do tamanho do *grid*. Se  $N = N_x$  tem-se então todas as variáveis armazenadas em RAM. Caso não seja possível armazenar todas as variáveis em RAM, será necessário definir uma matriz auxiliar de  $N_y \times N_z$ , para armazenar os valores do plano “i+1” no tempo “n” (usada no cálculo de H), nas interfaces entre os blocos de N planos.

- b)  $H [N][N_y][N_z]$  – N matrizes de  $N_y \times N_z$  elementos do tipo FIELD, sendo:
- $H [i][j][k].x$  – corresponde a  $E_x(j,k)$  do plano “i”
  - $H [i][j][k].y$  – corresponde a  $E_y(j,k)$  do plano “i”
  - $H [i][j][k].z$  – corresponde a  $E_z(j,k)$  do plano “i”

Valem as mesmas observações acima apresentadas para o campo elétrico. No caso do campo magnético, não sendo possível armazenar todas as variáveis em RAM, será necessário definir uma matriz auxiliar de  $N_y \times N_z$ , para armazenar os valores do plano “i-1” no tempo “n” (usada no cálculo de E), nas interfaces entre os blocos de N planos.

- c) Cel [N] [N<sub>y</sub>] [N<sub>z</sub>] – N matrizes de N<sub>y</sub> x N<sub>z</sub> elementos do tipo GRID, sendo:
- Cel [ i ][ j ][ k ].mat – índice “m” do material na célula (j,k) do plano “i”
  - Cel [ j ][ j ][ k ].fonte – indica se existe fonte na célula (j,k) do plano “i”

O valor de N pode ou não ser igual à N<sub>x</sub>, dependendo do tamanho do *grid*. No caso de N = N<sub>x</sub> tem-se então todas as variáveis armazenadas em RAM.

- d) Coef [ N<sub>m</sub> ] – vetor de N<sub>m</sub> elementos do tipo COEFIC, sendo:
- Coef [ m ].Ca – coeficiente de cálculo definido anteriormente
  - Coef [ m ].Cb – coeficiente de cálculo definido anteriormente
  - Coef [ m ].Da – coeficiente de cálculo definido anteriormente
  - Coef [ m ].Db – coeficiente de cálculo definido anteriormente

Observa-se que apenas os coeficientes são necessários para cálculo, sendo dispensável o armazenamento em RAM dos valores das propriedades.

- e) Fonte [ N<sub>f</sub> ] – vetor de N<sub>f</sub> elementos do tipo SOURCE, sendo:
- Fonte [ f ].x – coordenada “i” da célula onde a fonte “f” está presente;
  - Fonte [ f ].y – coordenada “j” da célula onde a fonte “f” está presente;
  - Fonte [ f ].z – coordenada “k” da célula a fonte “f” está presente;
  - Fonte [ f ].func – índice da expressão (seno, degrau, etc) para cálculo da fonte “f” presente na célula;
  - Fonte [ f ].dir – indica a orientação espacial (direção do campo)
  - Fonte [ f ].campo – tipo de fonte (campo E ou H) presente na célula;
  - Fonte [ f ].par1 – parâmetro da expressão de cálculo da fonte;
  - Fonte [ f ].par2 – parâmetro da expressão de cálculo da fonte;
  - Fonte [ f ].par3 – parâmetro da expressão de cálculo da fonte.

Por economia de memória, as propriedades eletromagnéticas da célula são acessadas de forma indireta, a partir do índice “m” do material, armazenados na variável Cel [ i ][ j ][ k ].mat.

Pelo mesmo motivo, os parâmetros das fontes presentes em uma célula são acessados de forma indireta. Para permitir que fossem associadas mais de uma fonte a uma determinada célula, a busca pela fonte é realizada pela comparação das coordenadas da fonte e da célula. Isto implica na busca em todo o vetor de fontes para identificar as fontes presentes na célula e então realizar o cálculo do valor. Como o número de fontes é pequeno, optou-se pela flexibilidade do algoritmo.

As fontes precisam ser calculadas a cada ciclo de tempo, pois dificilmente são constantes. Para otimização do processamento, a função de cálculo do valor das fontes só é chamada quando  $Cell[i][j][k].fonte = 1$ . Esse procedimento reduziu significativamente o tempo de processamento. Para fins ilustrativos, supondo um *grid* formado por 20 Matrizes de 20 x 20 elementos, com fonte presentes em duas células, o tempo de processamento foi reduzido em 70%, comparando-se com a situação na qual a função de cálculo das fontes é chamada para cada célula processada, embora todas as variáveis estivessem em memória RAM.

Para avaliar a aplicação da técnica proposta, é preciso modificar o algoritmo básico apresentado, a fim de permitir sua efetiva utilização. Com efeito, segue o algoritmo básico otimizado:

1. Pré-processamento:
  - 1.1. Definir as propriedades eletromagnéticas dos materiais envolvidos material
  - 1.2. Definir o tamanho do *grid* e o material presente em cada célula do *grid*
  - 1.3. Definir o instante de tempo final, quando será interrompido o processo de cálculo
2. Processamento inicial
  - 2.1. Definir os valores de  $\Delta$  e  $\Delta t$  (Eq. 29 e 30)
  - 2.2. Ler as propriedades eletromagnéticas dos materiais, calcular os respectivos coeficientes  $Ca_{i,j,k}$ ,  $Cb_{i,j,k}$ ,  $Da_{i,j,k}$  e  $Db_{i,j,k}$  (Eq. 25 a 28) e armazenar em memória (carregar vetor **Coef** para a RAM)
  - 2.3. Ler os parâmetros das fontes presentes no *grid* (carregar vetor **Fonte** para a RAM)
  - 2.4. Processar o *grid* e marcar as células com fontes presentes (definir **Cell**).
  - 2.5. Estabelecer os valores iniciais de campo elétrico e magnético em cada célula
3. Para cada plano do *grid*, a partir do primeiro (processamento do campo elétrico):
  - 3.1. Ler os índices “m” de material e os valores das células do plano “i”
  - 3.2. Ler os valores das componentes atuais de campo associadas ao plano “i”

- 3.3. Para cada célula (j,k) do plano i:
    - 3.3.1. Ler os valores pré-calculados de  $Ca_{i,j,k}$  e  $Cb_{i,j,k}$ , associados ao índice “m”
    - 3.3.2. Se existir fonte na célula, calcular o valor das fontes de campo elétrico presentes na célula, no instante n
    - 3.3.3. Calcular os valores das componentes de campo elétrico para o próximo instante n+1 (Eq. 33 a 35). Impor os valores das fontes de campo elétrico, quando aplicável
    - 3.3.4. Incrementar coordenadas. Se ainda não processadas todas as células do plano “i”, voltar para (3.3.1)
  - 3.4. Armazenar os valores calculados das componentes de campo elétrico
  - 3.5. Incrementar “i”. Se ainda não processadas todos os planos, voltar para (3.1)
4. Para cada plano do *grid*, a partir do primeiro (processamento do campo magnético):
    - 4.1. Ler os índices “m” de material e os valores das células do plano “i”
    - 4.2. Ler os valores das componentes atuais de campo associadas ao plano “i”. Os valores de campos elétricos usados são os valores para o instante n+1, já avaliados na etapa (3.3)
    - 4.3. Para cada célula (j,k) do plano “i”:
      - 4.3.1. Ler os valores pré-calculados de  $Da_{i,j,k}$  e  $Db_{i,j,k}$ , associados ao índice “m”
      - 4.3.2. Se existir fonte na célula, calcular o valor das fontes de campo magnético presentes na célula, no instante n
      - 4.3.3. Calcular os valores das componentes de campo magnético, para o próximo instante n+1 (Eq. 36 a 38). Impor os valores das fontes de campo magnético, quando aplicável
      - 4.3.4. Incrementar coordenadas. Se ainda não processadas todas as células do plano “i”, voltar para (4.3.1)
    - 4.4. Armazenar os valores calculados das componentes de campo magnético
    - 4.5. Incrementar “i”. Se ainda não processados todos os planos, voltar para (4.1)
5. Incrementar tempo. Caso não tenha alcançado o tempo final, voltar para (3).
  6. Apresentar resultados.

O espaço mínimo requerido de memória RAM é sumarizado na tabela abaixo.

Tab. 9 – Memória RAM necessária para processamento do algoritmo

Descrição	Memória RAM (bytes)
vetor das coeficientes associados aos materiais (sem armazenar as propriedades)	$16 N_m$
vetor dos tipos e parâmetros da fontes	$21 N_f$
matrizes associadas as características do <i>grid</i>	$2 N \times N_y \times N_z$
matrizes associadas aos campos elétrico e magnético	$24 (N+1) \times N_y \times N_z$

Assim, para o exemplo referido acima, considerando hipoteticamente  $N_m=256$  e  $N_f = 1000$ , seriam necessários 25.096 bytes para os vetores e 26.000.000 bytes para cada matriz. Na hipótese de  $N=1$ , implica num total aproximado de 49,6 MB, que é compatível com o equipamento alvo. Na hipótese de  $N=50$ , a memória mínima necessária passa para 1264,6 MB, o que atualmente é facilmente realizável em PCs de uso geral.

Com relação às operações de acesso a disco, o resultado da aplicação da técnica “organização de variáveis por plano espacial” é a drástica redução do número de acessos.

No início, existem duas operações de leitura, para carga dos vetores auxiliares Coef e Fonte, os quais permanecem em RAM durante o processo iterativo. No processo iterativo, para cada bloco de  $N$  planos são necessárias apenas 06 operações de leitura e 02 de escrita. Portanto, para cada interação de tempo, tem-se  $8N$  operações de acesso a disco, sendo  $N$  o número total blocos de matrizes no qual o *grid* é subdividido, conforme já apresentado.

Então, usando o exemplo do *grid* formado por 1000 matrizes de 1000 x 1000 células ( $10^9$  células), considerando  $N=50$ , tem-se no máximo 160 acessos a disco por interação. Se for considerado que, em cada operação leitura ou escrita em disco, seja demandado um tempo total de 10 ms (acesso e transferência de dados), tem-se apenas 1,6 s tomados com acesso a disco a cada interação de tempo, o que é plenamente aceitável. Esse tempo é uma pequena parcela do tempo necessário para a realização de todos os cálculos, estimados em 600 s por interação, com base nos resultados de simulação com *grids* menores (ver o capítulo de resultados).

Quando não é possível armazenar todas as variáveis em RAM, será necessário armazenar os dados de campo em todos os planos por pelo menos uma interação de tempo, para permitir o cálculo da interação seguinte. Dessa forma, para o exemplo antes referido seria necessário pelo menos 22,4 GB de espaço em disco. Embora esse espaço de disco esteja próximo ao especificado para o equipamento alvo, atualmente discos rígidos de 200 GB ou maiores são bastante comuns e acessíveis, os quais poderiam ser facilmente instalados.

No caso de ser desejado realizar uma animação, precisa-se ter a evolução dos valores de campo ao longo do tempo, o que impõe a utilização de maior espaço em disco. Mesmo nesses casos, ainda é possível executar o algoritmo em um único computador, até porque dificilmente o *grid* terá a dimensão de  $10^9$  células, tomada como exemplo extremo para avaliação das necessidades computacionais.

Considerando que o objeto das simulações é a interação de campo próximo, situadas a até  $1\lambda$  de distância da estrutura (10 a 20 células), uma hipótese mais realista é um *grid* formado por 200 matrizes de 200 x 200 células ( $8 \cdot 10^6$  células). Nesse caso, a necessidade de memória mínima é reduzida para algo em torno de 198,4 MB. Para *grids* dessa dimensão é possível rodar o algoritmo no PC alvo com todas as variáveis em memória RAM, sem grandes restrições.

### **6.3 Algoritmo completo do FDTD**

A adaptação do algoritmo básico otimizado, no qual está inserida a técnica denominada “organização de variáveis por plano espacial”, para incluir o tratamento das condições de borda (implementação da ABC) não é complexa.

Na verdade, para realizar a comparação entre simulações com e sem *UPML*, o programa implementa dois equacionamentos distintos, a ser executado conforme o valor do parâmetro *UPML*.

Assim, sendo escolhido um número de camadas absorventes igual a zero ( $UPML=0$ ), serão executados os cálculos conforme o algoritmo original de Yee. Caso contrário, será executado o algoritmo modificado com a técnica da *UPML*. As duas situações propostas e o equacionamento envolvido estão sumarizadas na seqüência.

#### **6.3.1 Implementação do algoritmo de Yee sem UPML**

A implementação do equacionamento original de Yee é realizada conforme algoritmo básico otimizado já apresentado.

As equações necessárias para o processamento foram reunidas abaixo, por conveniência:

a) definição dos incrementos  $\Delta$  e  $\Delta t$ :

$$N = \frac{\lambda}{\Delta} \geq 10 \quad (\text{Eq.29})$$

$$S = \frac{c \cdot \Delta t}{\Delta} = 0,5 \quad (\text{Eq.30})$$

onde:  $c$  – velocidade da luz no vácuo

b) cálculo do campo elétrico:

$$E_x|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot E_x|_{i,j,k}^n + Cb_{i,j,k} \left( H_z|_{i,j,k}^n - H_z|_{i,j-1,k}^n + H_y|_{i,j,k-1}^n - H_y|_{i,j,k}^n \right) \quad (\text{Eq.33})$$

$$E_y|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot E_y|_{i,j,k}^n + Cb_{i,j,k} \left( H_x|_{i,j,k}^n - H_x|_{i,j,k-1}^n + H_z|_{i-1,j,k}^n - H_z|_{i,j,k}^n \right) \quad (\text{Eq.34})$$

$$E_z|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot E_z|_{i,j,k}^n + Cb_{i,j,k} \left( H_y|_{i,j,k}^n - H_y|_{i-1,j,k}^n + H_x|_{i,j-1,k}^n - H_x|_{i,j,k}^n \right) \quad (\text{Eq.35})$$

$$Ca_{i,j,k} = \frac{1 - \frac{\sigma_{i,j,k} \cdot \Delta t}{2\epsilon_{i,j,k}}}{1 + \frac{\sigma_{i,j,k} \cdot \Delta t}{2\epsilon_{i,j,k}}} \quad (\text{Eq.25})$$

$$Cb_{i,j,k} = \frac{\frac{\Delta t}{\Delta \cdot \epsilon_{i,j,k}}}{1 + \frac{\sigma_{i,j,k} \cdot \Delta t}{2\epsilon_{i,j,k}}} \quad (\text{Eq.26})$$

c) cálculo do campo magnético:

$$H_x|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot H_x|_{i,j,k}^n + Db_{i,j,k} \left( E_y|_{i,j,k+1}^{n+1} - E_y|_{i,j,k}^{n+1} + E_z|_{i,j,k}^{n+1} - E_z|_{i,j+1,k}^{n+1} \right) \quad (\text{Eq.36})$$

$$H_y|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot H_y|_{i,j,k}^n + Db_{i,j,k} \left( E_z|_{i+1,j,k}^{n+1} - E_z|_{i,j,k}^{n+1} + E_x|_{i,j,k}^{n+1} - E_x|_{i,j,k+1}^{n+1} \right) \quad (\text{Eq.37})$$

$$H_z|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot H_z|_{i,j,k}^n + Db_{i,j,k} \left( E_x|_{i,j+1,k}^{n+1} - E_x|_{i,j,k}^{n+1} + E_y|_{i,j,k}^{n+1} - E_y|_{i+1,j,k}^{n+1} \right) \quad (\text{Eq.38})$$

$$Da_{i,j,k} = \frac{1 - \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}}{1 + \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}} \quad (\text{Eq.27})$$

$$Db_{i,j,k} = \frac{\frac{\Delta t}{\Delta \cdot \mu_{i,j,k}}}{1 + \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}} \quad (\text{Eq.28})$$

### 6.3.2 Implementação da UPML

Novamente, as equações necessárias para o processamento foram reunidas abaixo, por conveniência:

a) definição dos incrementos  $\Delta$  e  $\Delta t$ :

$$N = \frac{\lambda}{\Delta} \geq 10 \quad (\text{Eq.29})$$

$$S = \frac{c \cdot \Delta t}{\Delta} = 0,5 \quad (\text{Eq.30})$$

onde:  $c$  – velocidade da luz no vácuo

b) definição da UPML:

$$\sigma_w(w) = \sigma_{\max} \left( \frac{w}{d} \right)^m \quad (\text{Eq.64})$$

onde:

$w$  – distância percorrida no interior da UPML

$d = N_{upml} \cdot \Delta$  (a espessura da UPML adotada:  $5\Delta$  ou  $10\Delta$ )

$\sigma_w$  – refere-se à condutividade da UPML em uma das direções

$m = 4$

$$\sigma_{\max} = \frac{5}{\Delta \cdot \sqrt{\frac{\mu}{\epsilon}}} \quad (\text{a partir da Eq.65, Tab. 7 e valores adotados})$$

c) cálculo do campo elétrico:

$$F_x \Big|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot F_x \Big|_{i,j,k}^n + Cb_{i,j,k} \left( H_z \Big|_{i,j,k}^n - H_z \Big|_{i,j-1,k}^n + H_y \Big|_{i,j,k-1}^n - H_y \Big|_{i,j,k}^n \right) \quad (\text{Eq.104})$$

$$F_y \Big|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot F_y \Big|_{i,j,k}^n + Cb_{i,j,k} \left( H_x \Big|_{i,j,k}^n - H_x \Big|_{i,j,k-1}^n + H_z \Big|_{i-1,j,k}^n - H_z \Big|_{i,j,k}^n \right) \quad (\text{Eq.105})$$

$$F_z \Big|_{i,j,k}^{n+1} = Ca_{i,j,k} \cdot F_z \Big|_{i,j,k}^n + Cb_{i,j,k} \left( H_y \Big|_{i,j,k}^n - H_y \Big|_{i-1,j,k}^n + H_x \Big|_{i,j-1,k}^n - H_x \Big|_{i,j,k}^n \right) \quad (\text{Eq.106})$$

$$G_x \Big|_{i,j,k}^{n+1} = Kd \Big|_{i,j,k}^y \cdot G_x \Big|_{i,j,k}^n + Ke \Big|_{i,j,k}^y \cdot \left( F_x \Big|_{i,j,k}^{n+1} - F_x \Big|_{i,j,k}^n \right) \quad (\text{Eq.107})$$

$$G_y \Big|_{i,j,k}^{n+1} = Kd \Big|_{i,j,k}^z \cdot G_y \Big|_{i,j,k}^n + Ke \Big|_{i,j,k}^z \cdot \left( F_y \Big|_{i,j,k}^{n+1} - F_y \Big|_{i,j,k}^n \right) \quad (\text{Eq.108})$$

$$G_z \Big|_{i,j,k}^{n+1} = Kd \Big|_{i,j,k}^x \cdot G_z \Big|_{i,j,k}^n + Ke \Big|_{i,j,k}^x \cdot \left( F_z \Big|_{i,j,k}^{n+1} - F_z \Big|_{i,j,k}^n \right) \quad (\text{Eq.109})$$

$$E_x \Big|_{i,j,k}^{n+1} = Ka \Big|_{i,j,k}^z \cdot E_x \Big|_{i,j,k}^n + Kb \Big|_{i,j,k}^{x,z} \cdot G_x \Big|_{i,j,k}^{n+1} - Kc \Big|_{i,j,k}^{x,z} \cdot G_x \Big|_{i,j,k}^n \quad (\text{Eq.110})$$

$$E_y \Big|_{i,j,k}^{n+1} = Ka \Big|_{i,j,k}^x \cdot E_y \Big|_{i,j,k}^n + Kb \Big|_{i,j,k}^{y,x} \cdot G_y \Big|_{i,j,k}^{n+1} - Kc \Big|_{i,j,k}^{y,x} \cdot G_y \Big|_{i,j,k}^n \quad (\text{Eq.111})$$

$$E_z \Big|_{i,j,k}^{n+1} = Ka \Big|_{i,j,k}^y \cdot E_z \Big|_{i,j,k}^n + Kb \Big|_{i,j,k}^{z,y} \cdot G_z \Big|_{i,j,k}^{n+1} - Kc \Big|_{i,j,k}^{z,y} \cdot G_z \Big|_{i,j,k}^n \quad (\text{Eq.112})$$

$$Ca_{i,j,k} = \frac{1 - \frac{\sigma_{i,j,k} \cdot \Delta t}{2\mathcal{E}_{i,j,k}}}{1 + \frac{\sigma_{i,j,k} \cdot \Delta t}{2\mathcal{E}_{i,j,k}}} \quad (\text{Eq.25})$$

$$Cb_{i,j,k} = \frac{\frac{\Delta t}{\Delta \cdot \mathcal{E}_{i,j,k}}}{1 + \frac{\sigma_{i,j,k} \cdot \Delta t}{2\mathcal{E}_{i,j,k}}} \quad (\text{Eq.26})$$

$$Ka \Big|_{i,j,k}^w = \frac{1 - \frac{\sigma_w \Delta t}{2\mathcal{E}}}{1 + \frac{\sigma_w \Delta t}{2\mathcal{E}}} \quad (\text{Eq.113})$$

$$Kb \Big|_{i,j,k}^{v,w} = \frac{1 + \frac{\sigma_v \Delta t}{2\mathcal{E}}}{1 + \frac{\sigma_w \Delta t}{2\mathcal{E}}} \quad (\text{Eq.114})$$

$$Kc \Big|_{i,j,k}^{v,w} = \frac{1 - \frac{\sigma_v \Delta t}{2\mathcal{E}}}{1 + \frac{\sigma_w \Delta t}{2\mathcal{E}}} \quad (\text{Eq.115})$$

$$Kd \Big|_{i,j,k}^w = \frac{1 - \frac{\sigma_w \Delta t}{2\mathcal{E}}}{1 + \frac{\sigma_w \Delta t}{2\mathcal{E}}} \quad (\text{Eq.116})$$

$$Ke \Big|_{i,j,k}^w = \frac{1}{1 + \frac{\sigma_w \Delta t}{2\mathcal{E}}} \quad (\text{Eq.117})$$

onde:

$v$  e  $w$  referem-se às direções  $x$ ,  $y$  ou  $z$

em  $\sigma_v$  e  $\sigma_w$  foram omitidos os índices  $i, j, k$  que identificam a célula

d) cálculo do campo magnético:

$$P_x|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot P_x|_{i,j,k}^n + Db_{i,j,k} \left( E_y|_{i,j,k+1}^{n+1} - E_y|_{i,j,k}^{n+1} + E_z|_{i,j,k}^{n+1} - E_z|_{i,j+1,k}^{n+1} \right) \quad (\text{Eq.118})$$

$$P_y|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot P_y|_{i,j,k}^n + Db_{i,j,k} \left( E_z|_{i+1,j,k}^{n+1} - E_z|_{i,j,k}^{n+1} + E_x|_{i,j,k}^{n+1} - E_x|_{i,j,k+1}^{n+1} \right) \quad (\text{Eq.119})$$

$$P_z|_{i,j,k}^{n+1} = Da_{i,j,k} \cdot P_z|_{i,j,k}^n + Db_{i,j,k} \left( E_x|_{i,j+1,k}^{n+1} - E_x|_{i,j,k}^{n+1} + E_y|_{i,j,k}^{n+1} - E_y|_{i+1,j,k}^{n+1} \right) \quad (\text{Eq.120})$$

$$Q_x|_{i,j,k}^{n+1} = Ld|_{i,j,k}^y \cdot Q_x|_{i,j,k}^n + Le|_{i,j,k}^y \cdot \left( P_x|_{i,j,k}^{n+1} - P_x|_{i,j,k}^n \right) \quad (\text{Eq.121})$$

$$Q_y|_{i,j,k}^{n+1} = Ld|_{i,j,k}^z \cdot Q_y|_{i,j,k}^n + Le|_{i,j,k}^z \cdot \left( P_y|_{i,j,k}^{n+1} - P_y|_{i,j,k}^n \right) \quad (\text{Eq.122})$$

$$Q_z|_{i,j,k}^{n+1} = Ld|_{i,j,k}^x \cdot Q_z|_{i,j,k}^n + Le|_{i,j,k}^x \cdot \left( P_z|_{i,j,k}^{n+1} - P_z|_{i,j,k}^n \right) \quad (\text{Eq.123})$$

$$H_x|_{i,j,k}^{n+1} = La|_{i,j,k}^z \cdot H_x|_{i,j,k}^n + Lb|_{i,j,k}^{x,z} \cdot Q_x|_{i,j,k}^{n+1} - Lc|_{i,j,k}^{x,z} \cdot Q_x|_{i,j,k}^n \quad (\text{Eq.124})$$

$$H_y|_{i,j,k}^{n+1} = La|_{i,j,k}^x \cdot H_y|_{i,j,k}^n + Lb|_{i,j,k}^{y,x} \cdot Q_y|_{i,j,k}^{n+1} - Lc|_{i,j,k}^{y,x} \cdot Q_y|_{i,j,k}^n \quad (\text{Eq.125})$$

$$H_z|_{i,j,k}^{n+1} = La|_{i,j,k}^y \cdot H_z|_{i,j,k}^n + Lb|_{i,j,k}^{z,y} \cdot Q_z|_{i,j,k}^{n+1} - Lc|_{i,j,k}^{z,y} \cdot Q_z|_{i,j,k}^n \quad (\text{Eq.126})$$

$$Da_{i,j,k} = \frac{1 - \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}}{1 + \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}} \quad (\text{Eq.27})$$

$$Db_{i,j,k} = \frac{\Delta \cdot \mu_{i,j,k}}{1 + \frac{\sigma_{i,j,k}^* \cdot \Delta t}{2\mu_{i,j,k}}} \quad (\text{Eq.28})$$

$$La|_{i,j,k}^w = \frac{1 - \frac{\sigma_w \Delta t}{2\mu}}{1 + \frac{\sigma_w \Delta t}{2\mu}} \quad (\text{Eq.127})$$

$$Lb|_{i,j,k}^{v,w} = \frac{1 + \frac{\sigma_v \Delta t}{2\mu}}{1 + \frac{\sigma_w \Delta t}{2\mu}} \quad (\text{Eq.128})$$

$$Lc|_{i,j,k}^{v,w} = \frac{1 - \frac{\sigma_v \Delta t}{2\mu}}{1 + \frac{\sigma_w \Delta t}{2\mu}} \quad (\text{Eq.129})$$

$$Ld|_{i,j,k}^w = \frac{1 - \frac{\sigma_w \Delta t}{2\mu}}{1 + \frac{\sigma_w \Delta t}{2\mu}} \quad (\text{Eq.130})$$

$$Le|_{i,j,k}^w = \frac{1}{1 + \frac{\sigma_w \Delta t}{2\mu}} \quad (\text{Eq.131})$$

onde:

$v$  e  $w$  referem-se às direções  $x$ ,  $y$  ou  $z$

em  $\sigma_v$  e  $\sigma_w$  foram omitidos os índices  $i,j,k$  que identificam a célula

Com base na formulação desenvolvida para implementação da *UPML*, se fazem necessárias as alterações apresentadas a seguir.

Após a estender o *grid* para a *UPML*, os coeficientes das expressões ainda podem ser calculados previamente e acessados pelo índice do material. Mas no interior da *UPML* as propriedades também variam com a posição, sendo necessário armazenar essas variações possíveis. Assim sendo, a busca dos coeficientes não depende apenas do material da célula, mas também da posição.

Percebe-se isso ao analisar as equações que definem  $K$ ,  $L$  e  $\sigma_w$ , o que implicará em na utilização de vetores e matrizes internos a estrutura que define os elementos do tipo COEFIC, com dimensões do número de células da *UPML* + 1. Embora isso represente um pequeno aumento da ocupação de memória, é preferível realizar o cálculo prévio dos coeficientes para reduzir o número de operações ao estritamente necessário.

Para atender a formulação da UPML, precisa-se alocar área de memória para as quatro variáveis auxiliares de campo, cujo valor calculado também precisa ser armazenado para a interação seguinte. Adicionalmente, às etapas (3.3.3) e (4.3.3) de cálculo das componentes de campo são desdobradas, cada uma, em três etapas de cálculo consecutivas, justamente para atualização dos valores das variáveis auxiliares de campo. Isso está ilustrado abaixo:

- 3.3.3.A. Calcular os valores das componentes de campo auxiliar F, para o próximo instante de tempo  $n+1$  (Eq. 104 a 106)
- 3.3.3.B. Calcular os valores das componentes de campo auxiliar G, para o próximo instante de tempo  $n+1$  (Eq. 107 a 109)
- 3.3.3.C. Calcular os valores das componentes de campo elétrico, para o próximo instante de tempo  $n+1$  (Eq. 110 a 112), adicionando vetorialmente os valores da fonte
  
- 4.3.3.A. Calcular os valores das componentes de campo auxiliar P, para o próximo instante de tempo  $n+1$  (Eq. 118 a 120)
- 4.3.3.B. Calcular os valores das componentes de campo auxiliar Q, para o próximo instante de tempo  $n+1$  (Eq. 121 a 123)
- 4.3.3.C. Calcular os valores das componentes de campo magnético, para o próximo instante de tempo  $n+1$  (Eq. 124 a 126), adicionando vetorialmente os valores da fonte

Percebe-se que, associado aos excelentes resultados obtidos, virtualmente eliminando os problemas de reflexão numérica, a utilização da UPML impõe um aumento no custo computacional:

- a) tem-se maior número de células no *grid*, devido a camada externa extra formada pelo meio absorvente;
- b) para o cálculo de cada componente de campo, é necessário o cálculo prévio de duas variáveis auxiliares, cujo resultado também precisa ser armazenada para as interações futuras;
- c) tem-se aumentado o número de coeficientes por célula (de dois para sete para cada variável de campo)

Para fins comparativos, será avaliada a implementação do algoritmo, sem e com UPML, em um espaço definido por *grid* de 1000 planos de 1000x1000 células e uma UPML de 5 células de espessura ( $N_{upml} = l = 5$ ). Os números comparativos são apresentados nas tabelas seguintes:

Tab. 10 – Número de células

	Yee	Yee + UPML
Células por Plano	$N_y \times N_z$	$(N_y+2l) \times (N_z+2l)$
Células no <i>grid</i> ( $N_{cel}$ )	$N_x \times N_y \times N_z$	$(N_x+2l) \times (N_y+2l) \times (N_z+2l)$
Total de células do exemplo	1.000.000.000	1.030.301.000

Tab. 11 – Memória RAM associada ao *grid*

Descrição	Yee	Yee + UPML
Memória associada a um plano do <i>grid</i>	$2 N_y \times N_z$	$2 (N_y+2l) \times (N_z+2l)$
Total de RAM (MB/plano)	1,91	1,95

Tab. 12 – Memória RAM e espaço mínimo em disco associados aos campos

Descrição	Yee	Yee + UPML
Memória associada aos campos (para cada plano do <i>grid</i> )	$24 N_y \times N_z$	$72 (N_y+2l) \times (N_z+2l)$
Total de RAM (MB/plano)	22,9	70,0
Espaço mínimo em disco (GB)	22,4	69,1

Por hipótese, não é possível armazenar todas as variáveis na RAM (pior caso)

Tab. 13 – Operações de cálculo matemático das componentes de campo

Descrição	Yee	Yee + UPML
Adições	$24N_{cel}$	$48N_{cel}$
Multiplicações	$12N_{cel}$	$42N_{cel}$

Apenas operações para avaliação das componentes durante cálculo iterativo

Portanto, ao adotar a UPML conclui-se que:

- o número de bytes para armazenamento das características do *grid* aumenta em 2%;
- o número de bytes para armazenamento em disco dos valores de campo aumenta em 208%;
- a quantidade de memória RAM utilizada para processamento dos valores de campo aumenta 205%;
- o número de operações de adição aumentam em 100% e as operações de multiplicação aumentam em 250%

Apesar do aumento significativo dos recursos computacionais exigidos, o algoritmo empregando técnica da *UPML* é possível de ser utilizado no equipamento objeto desse trabalho.

## 7. Resultados

A título de experimentação, foram desenvolvidos programas em linguagem C que implementaram os algoritmos propostos:

- algoritmo de Yee (sem ABC)
- algoritmo de Yee + *UPML*

Nas simulações foram utilizadas as seguintes funções de fonte de excitação:

a) pulso Gaussiano – para excitação transiente.

$$U(n) = A \cdot e^{-\left(\frac{n-60}{12}\right)^2} \quad (\text{Eq. 132})$$

onde:

$U(n)$  – representa qualquer um dos campos

$A$  – representa o valor de pico sinal

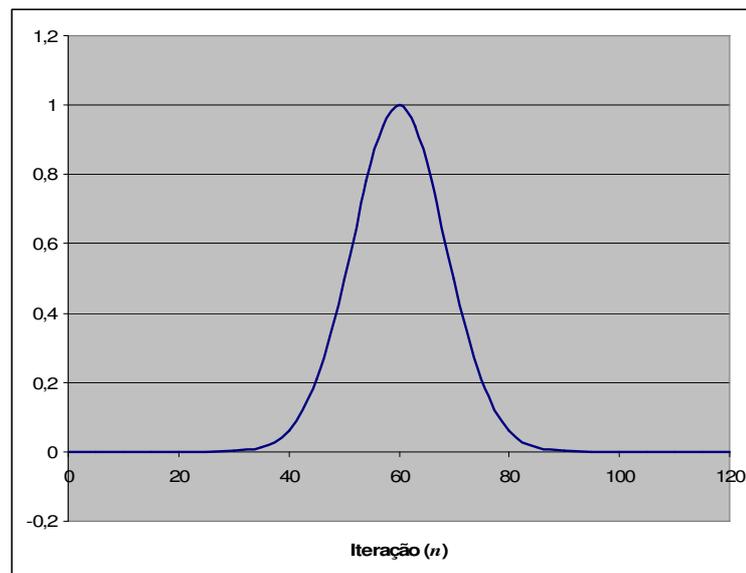


Fig. 4 – Gráfico da função pulso Gaussiano para valor de pico unitário ( $A = 1$ )

b) seno – nas simulações com excitações permanentes

$$U(n) = A \cdot \sin(2\pi \cdot f \cdot n \cdot \Delta t) \quad (\text{Eq. 133})$$

onde:

$U(n)$  – representa qualquer um dos campos

$f$  – representa a frequência do sinal senoidal, em Hz

$A$  – representa o valor de pico sinal

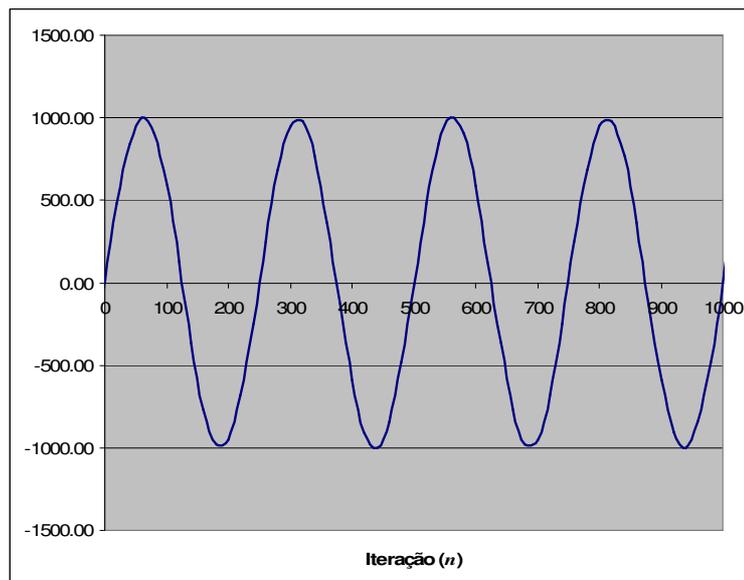


Fig. 5 – Gráfico da função seno utilizada nos ensaios

O gráfico acima, considera a frequência  $f = 2.4\text{GHz}$  ( $\lambda_{\text{wave}}=125\text{mm}$ ), o valor de pico igual a  $A=1000$ . Ainda, foram definidos o comprimento de onda de corte do *grid*  $\lambda=20\text{mm}$ ,  $N=20$  e  $S=0,5$ . Dessa forma, tem-se  $\Delta t = 1,67\text{ps}$  e  $\Delta = 1\text{mm}$ .

Caso não explicitamente mencionado em contrário, as fontes de excitação foram posicionadas no centro do *grid*, ocupando uma célula de extensão.

## 7.1 Tempo de processamento

Embora o método FDTD seja divulgado como um processo de simulação lento e de grande exigência computacional, os resultados preliminares obtidos mostraram que a essa opinião já pode ser modificada para situações mais simples. A tabela abaixo sumariza o tempo de processamento, em função das dimensões do *grid*, para as técnicas de ABC implementadas e rodando em um PC, conforme descrito nesse trabalho. Os valores abaixo referem-se ao tempo necessário para serem processadas 300 iterações de tempo.

Tab. 14 – Tempo de processamento em função das dimensões do *grid* e algoritmo

Dimensões do <i>grid</i>	Yee	Yee + UPML
50 x 50 x 50	34 s	47 s
100 x 100 x 100	230 s	268 s
150 x 150 x 150	714 s	871 s

Tempo necessário para o processamento de 300 interações de tempo

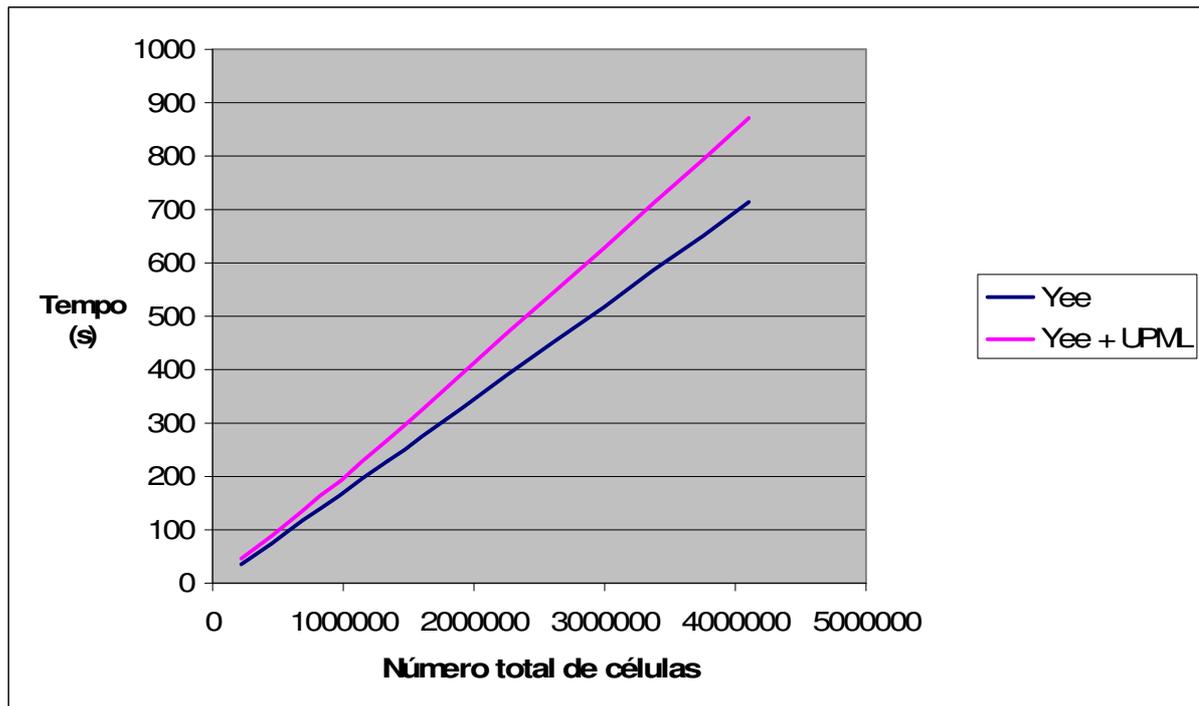


Fig. 6 – Gráfico tempo de processamento x tamanho do grid

Para as dimensões utilizadas, todas as variáveis foram armazenadas em RAM. A camada de meio absorvente era formada por 5 células.

Com base nos resultados, percebe-se que:

- o algoritmo sem *UPML* é mais rápido;
- o tempo de execução aumenta linearmente com o número de células, evidenciando a interferência direta do número de operações no tempo de processamento.

## 7.2 Otimização do acesso ao disco

Para comprovar a otimização do acesso a disco em função da técnica de “organização de variáveis por plano espacial”, o programa foi executado em uma máquina com recursos limitados de memória RAM, para um grid de dimensões suficientemente elevadas para haver a necessidade de uso da cache de disco.

Para tanto, o programa que implementou algoritmo de Yee + *UPML* foi executado em um equipamento com 256MB e sistema operacional Windows XP. O tamanho do *grid* era 150x150x150 células.

Como a idéia é minimizar o acesso a disco, optou-se por limitar a alocação de memória realizada pelo programa ao tamanho da RAM realmente disponível. E, com essa finalidade, o algoritmo foi desenvolvido de tal forma a permitir a escolha do número de planos a serem lidos do disco para a RAM, processados e novamente armazenados em disco.

Se escolhido número de planos igual ao número de matrizes que compõe o *grid*, resulta no não gerenciamento do acesso a disco pelo programa. É realizada a alocação de toda a memória necessária como se houvesse espaço suficiente em RAM, cabendo ao sistema operacional o gerenciamento com a utilização da cache de disco.

Os resultados estão expressos na tabela abaixo:

Tab. 15 – Tempo de processamento com e sem gerenciamento de memória pelo programa

Gerenciamento realizado por	Tempo para 10 interações
Sistema operacional	744
Programa	638

As seguintes conclusões são possíveis:

- quando há necessidade de acesso a disco para rodar o algoritmo, esse torna-se significativamente mais lento (os resultados acima são para dez interações e os da Tabela 14 são para 300 interações);
- quando o gerenciamento de acesso a disco é realizado pelo programa, é possível uma redução de pelo menos 15% no tempo de execução.

### **7.3 Ensaios de propagação e reflexão da onda nas bordas**

Foram adotados  $N=20$  e  $S=0,5$  para todas as simulações, sendo alterado o comprimento de onda de corte do *grid* ( $\lambda$ ), conforme o valor de  $\Delta$  desejado.

Por razões de simetria, foi monitorada a evolução da intensidade do valor de campo elétrico resultante ao longo do tempo, nas seguintes células:

Tab. 16 – Células para monitoramento dos valores de campo

Descrição do ponto	Coordenadas ( i , j, k )
P0	$N_x/2 - 1, N_y/2 - 1, N_z/2 - 1$
P1	$N_x/2 - 1, N_y/2 - 1, N_z/4 - 1$
P2	$N_x/2 - 1, N_y/2 - 1, 3N_z/4 - 1$
P3	$N_x/2 - 1, N_y/2 - 1, 5$
P4	$N_x/2 - 1, N_y/2 - 1, N_z - 6$
P5	$N_x/2 - 1, N_y/4 - 1, N_z/2 - 1$
P6	$N_x/2 - 1, 3N_y/4 - 1, N_z/2 - 1$
P7	$N_x/2 - 1, 5, N_z/2 - 1$
P8	$N_x/2 - 1, N_y - 6, N_z/2 - 1$

Coordenadas internas, referindo-se ao grid sem a camada absorvente

A figuras abaixo ilustram a localização do plano e dos pontos de amostragem.

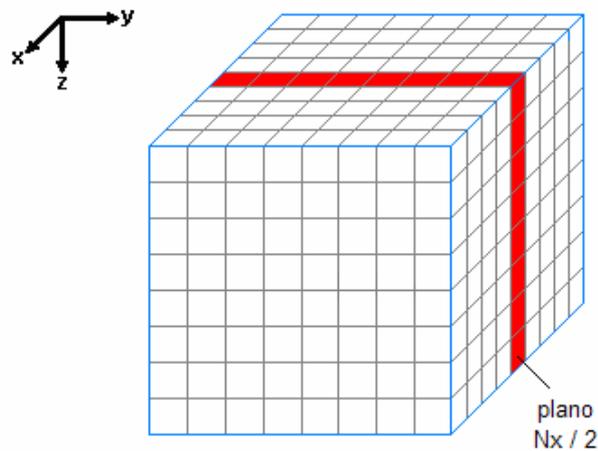


Fig. 7 – Plano onde estão contidos os pontos de amostragem

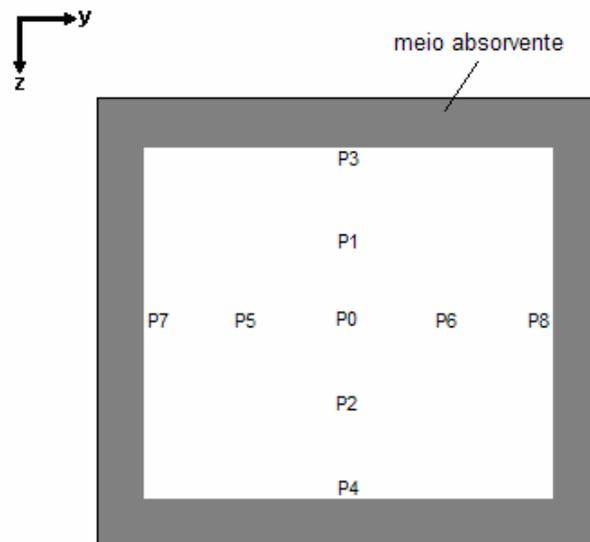


Fig. 8 – Pontos de amostragem dos valores de campo

Montou-se a simulação de um dipolo curto no vácuo, excitado de campo elétrico com excitação por uma fonte senoidal, centrada no dipolo e orientada na direção do eixo z. O dipolo curto está contido e centrado no plano  $N_x / 2$ , ilustrado na figura 7 acima, cuja orientação e dimensões estão detalhadas na figura abaixo:

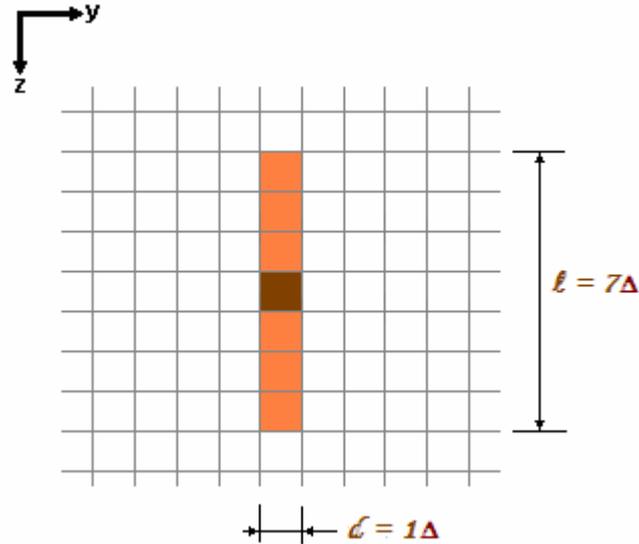


Fig. 45 – Detalhe da orientação e dimensões do dipolo curto

Considerando o comprimento de onda da fonte ( $\lambda_{\text{wave}} = 125\text{mm}$ ), têm-se as seguintes relações para o dipolo curto objeto de simulação ( $\Delta$  em mm):

$$\frac{l}{\lambda_{\text{wave}}} = 0,056\Delta \quad \text{e} \quad \frac{d}{\lambda_{\text{wave}}} = 0,008\Delta$$

Nas simulações em que foi usado um *grid* cúbico (mesmo número de células em todas as direções), foi monitorada a intensidade do campo elétrico resultante, calculada a partir do valor das componentes nas direções x, y e z, conforme segue:

$$E = \left( E_x^2 + E_y^2 + E_z^2 \right)^{1/2} \quad (\text{Eq. 134})$$

#### a) Simulação do Algoritmo de Yee sem UPML

O tamanho do *grid* utilizado foi de 100x100x100 células. Percebeu-se que, havendo o número de interações de tempo suficiente, a onda propaga-se até as bordas do *grid*, refletindo para o interior do mesmo, provocando oscilações inexistentes nos valores de campo. Os resultados típicos são apresentados abaixo.

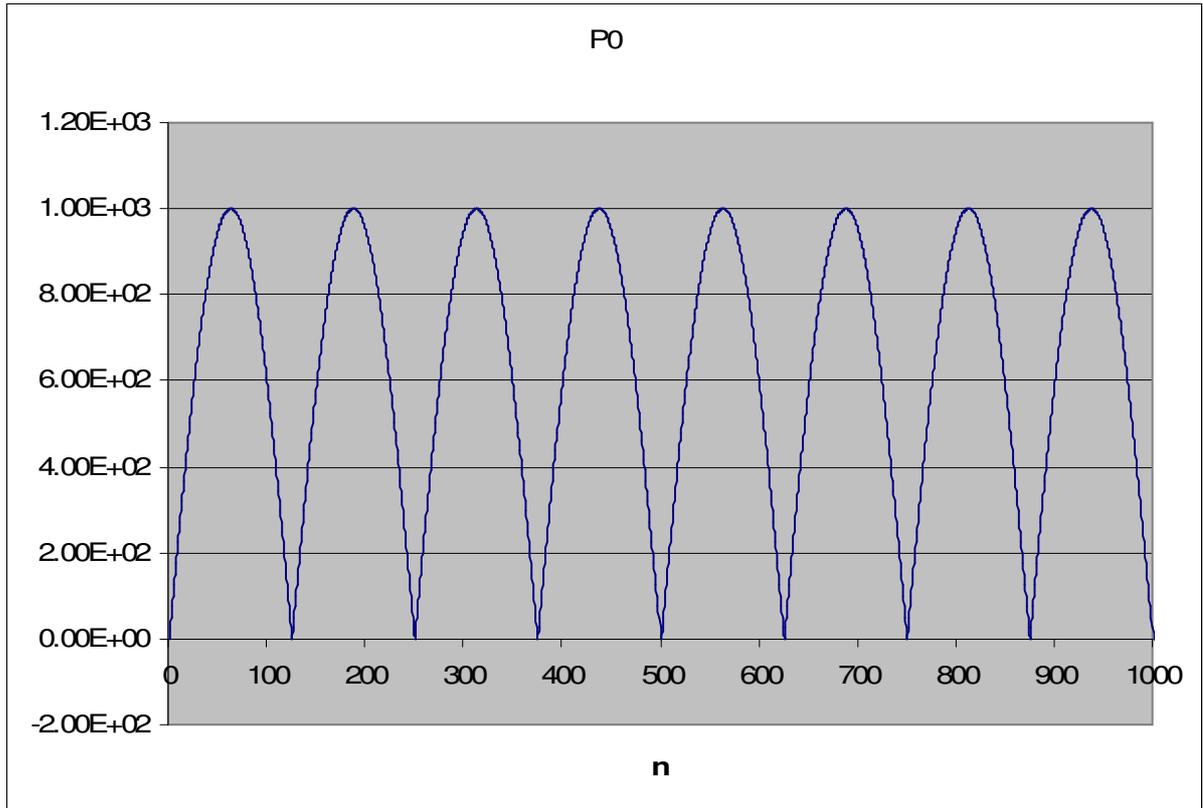


Fig. 9 – Simulação 1 ponto P0, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

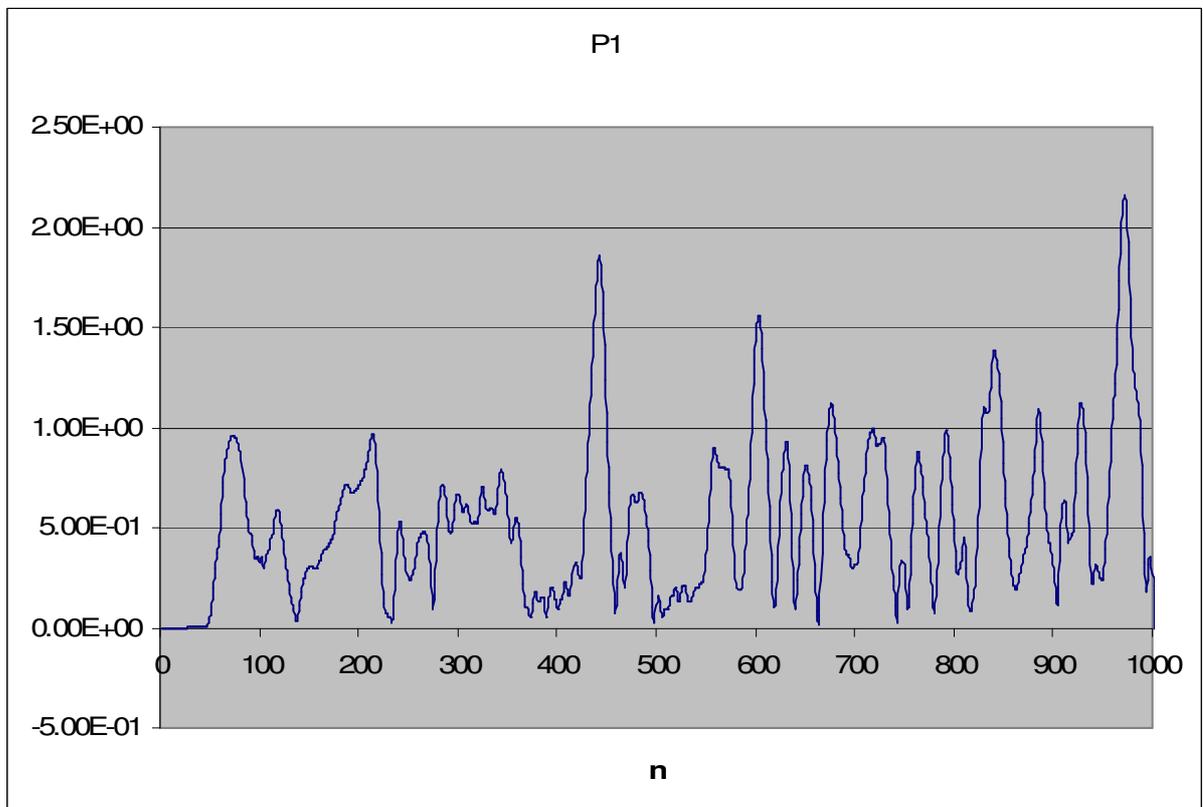


Fig. 10 – Simulação 1 ponto P1, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

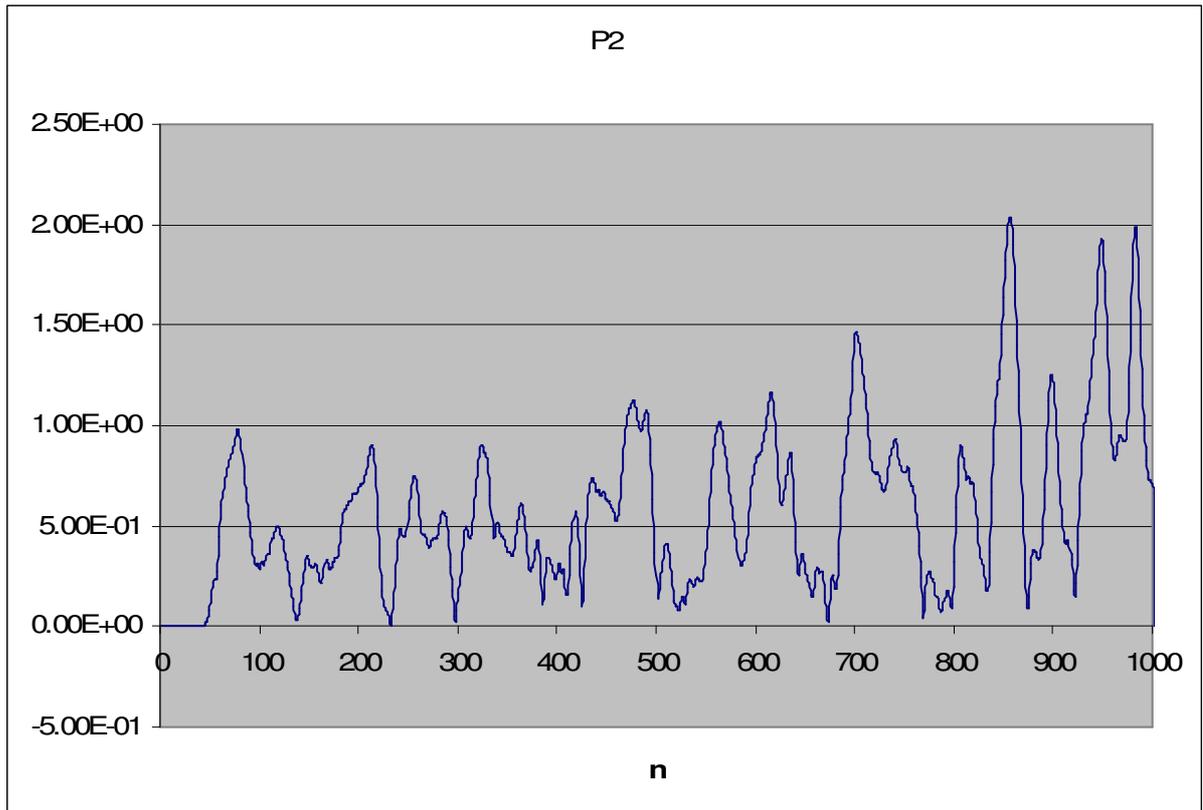


Fig. 11 – Simulação 1 ponto P2, , campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

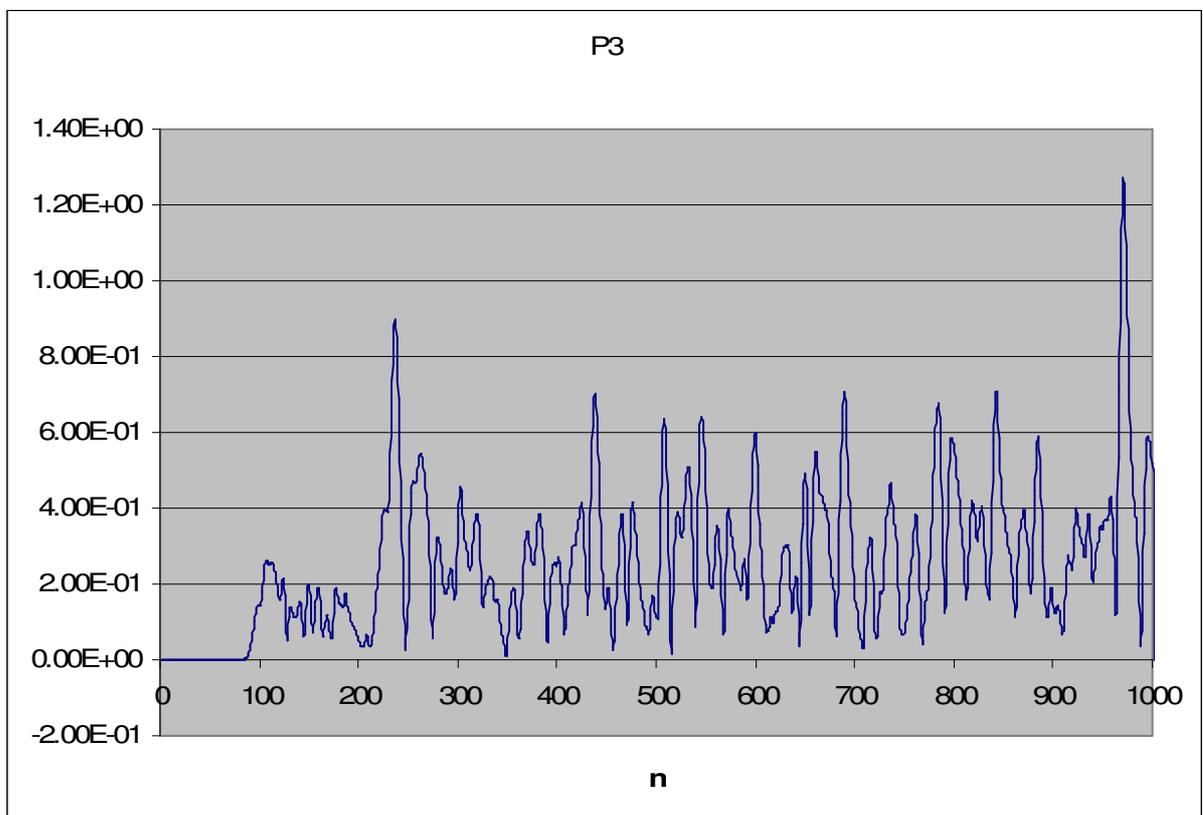


Fig. 12 – Simulação 1 ponto P3, , campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

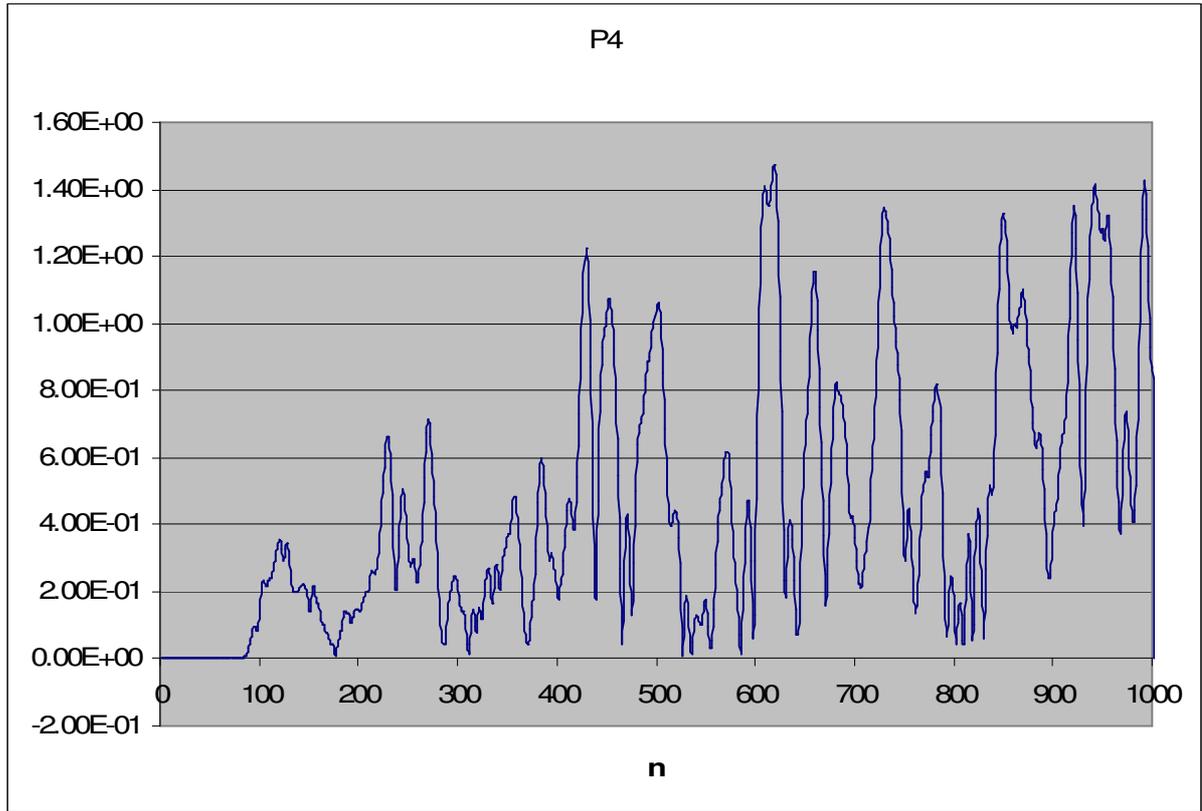


Fig. 13 – Simulação 1 ponto P4, , campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

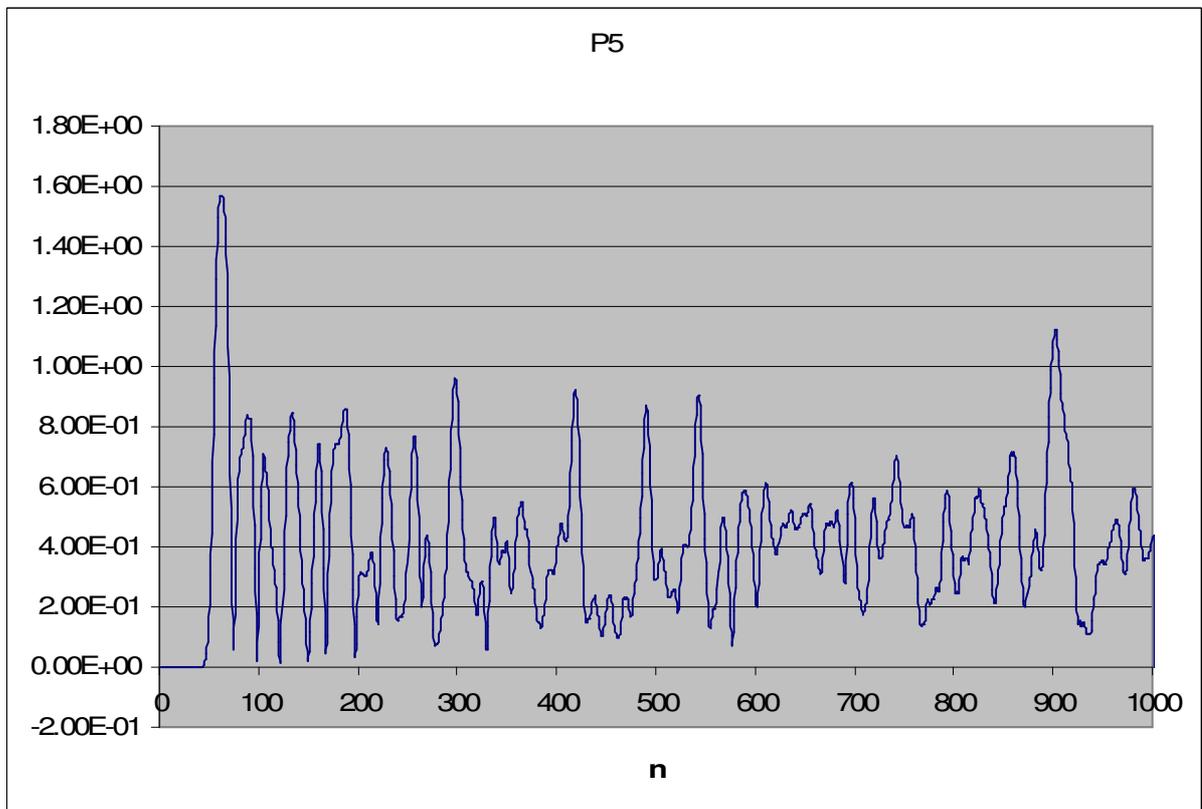


Fig. 14 – Simulação 1 ponto P5, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

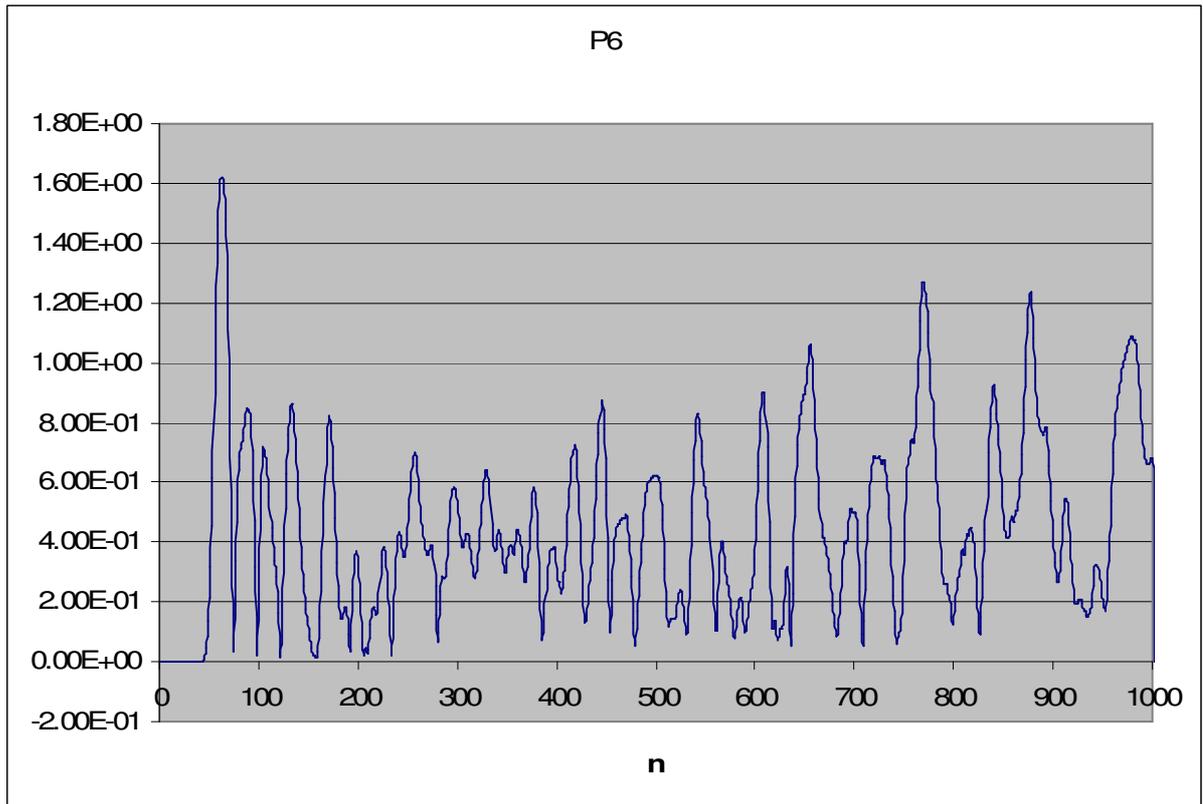


Fig. 15 – Simulação 1 ponto P6, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

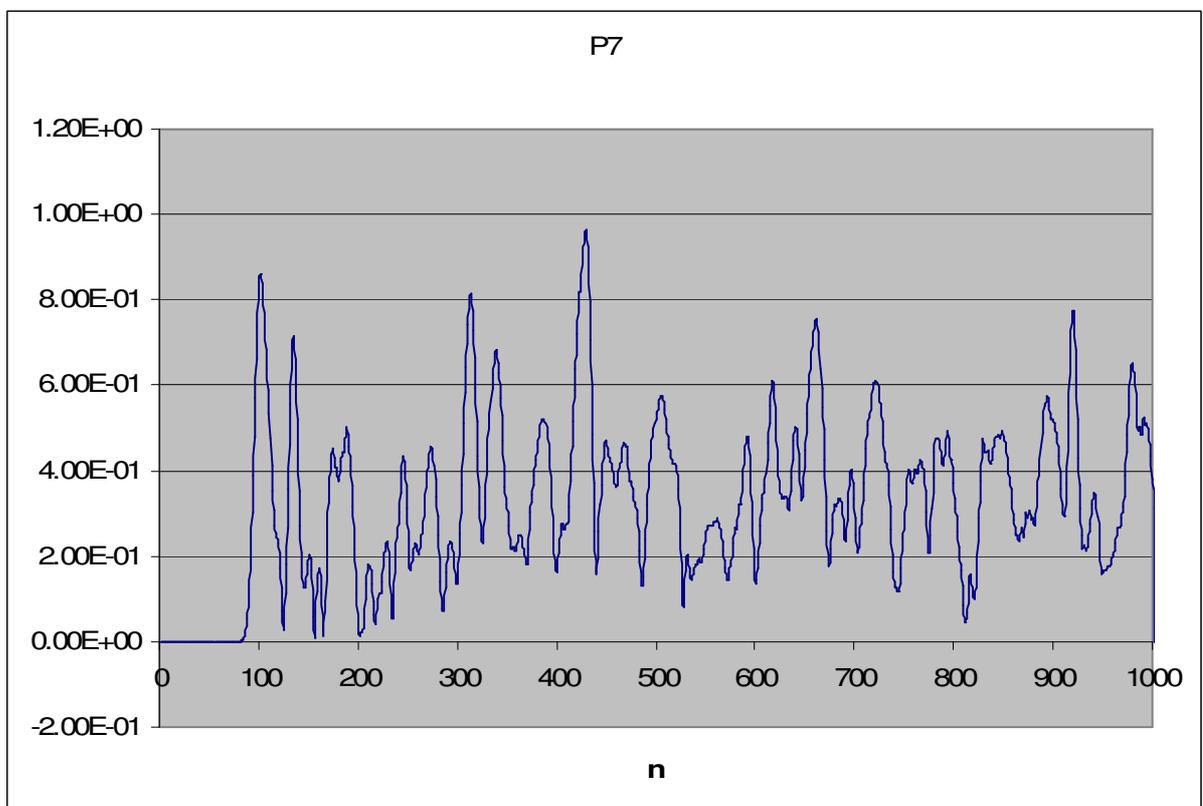


Fig. 16 – Simulação 1 ponto P7, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

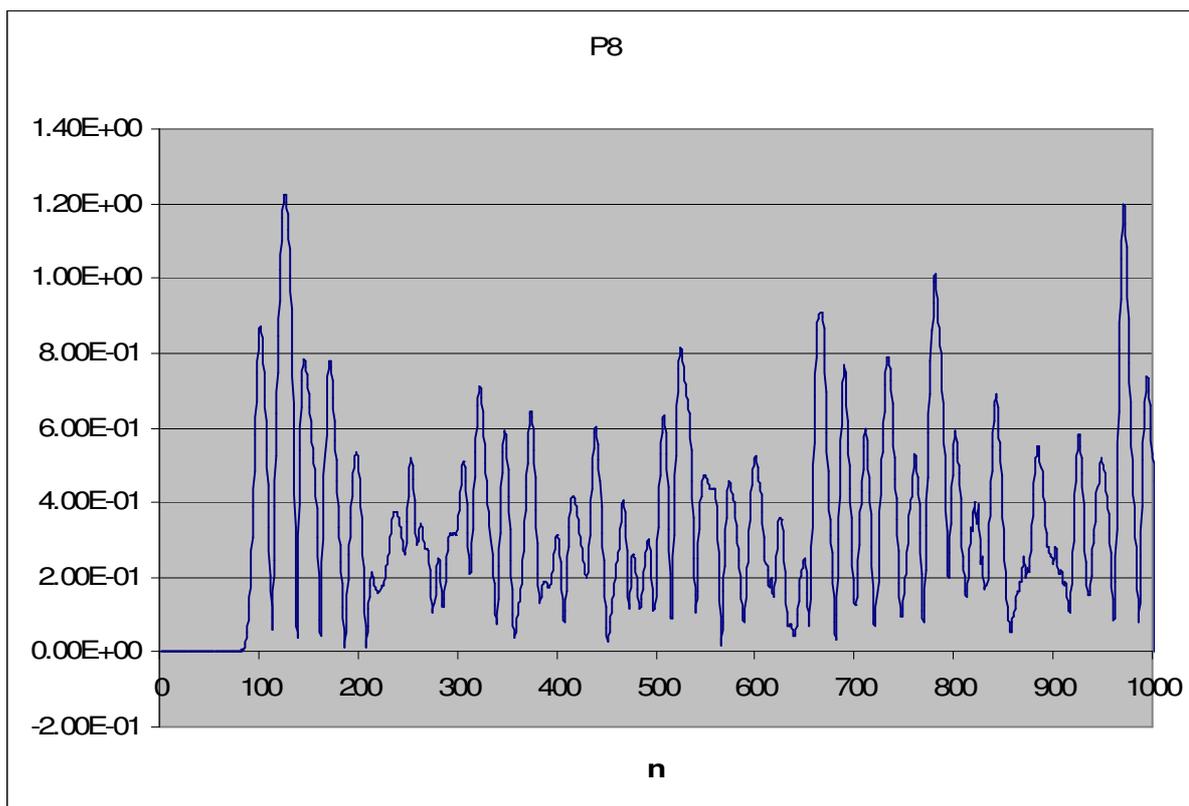


Fig. 17 – Simulação 1 ponto P8, , campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

Na tentativa de avaliar o efeito da amostragem sobre os resultados obtidos com o algoritmo sem a técnica da *UPML*, adotou-se um valor de  $\lambda=5\text{mm}$  (5 vezes menor que o valor anterior). O número de células do *grid* foi mantido (100x100x100 células).

Dessa forma, a relação entre o comprimento de onda da fonte ( $\lambda_{\text{wave}}$ ) e o comprimento de onda de corte do *grid* ( $\lambda$ ) passou de 6,25 para 25.

Não ficou evidente se houve ou não um melhor desempenho do algoritmo sem *UPML* nessa situação, pois as oscilações dos valores de campo permaneceram significativas a ponto de não ser possível perceber a forma de onda. Os resultados típicos são apresentados na seqüência.

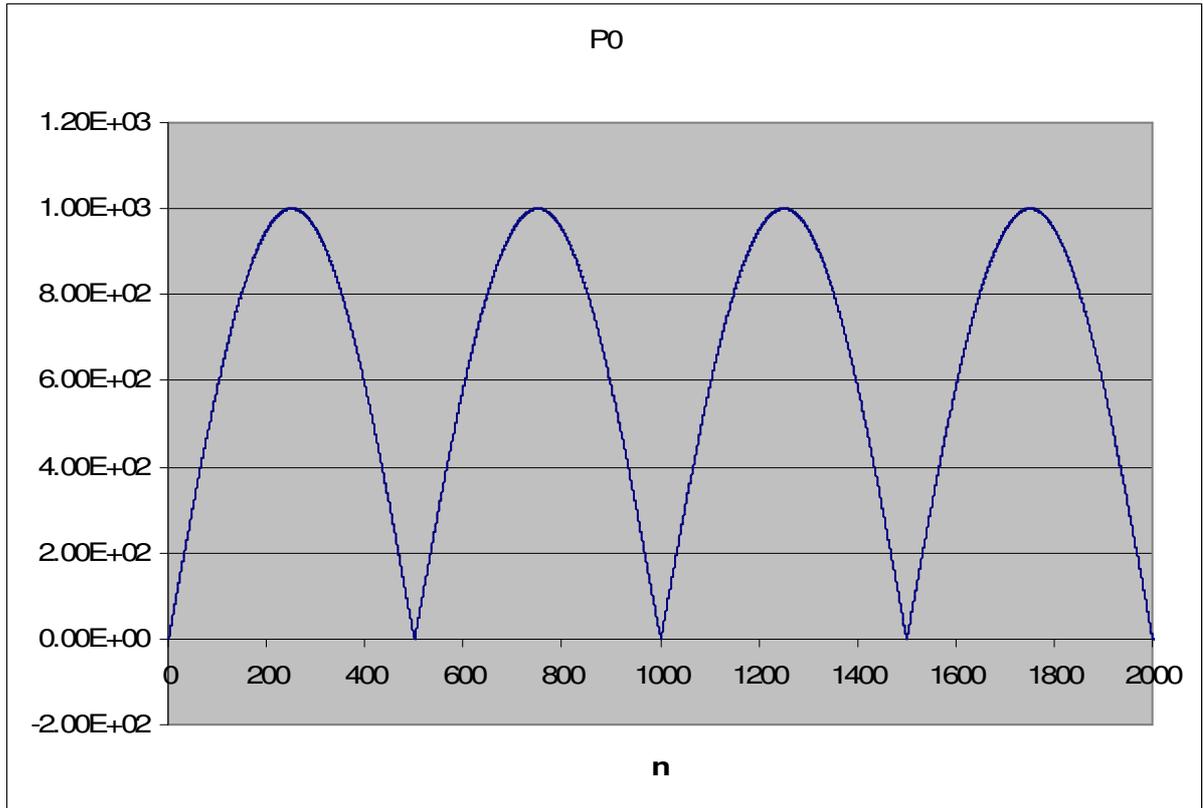


Fig. 18 – Simulação 2 ponto P0, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

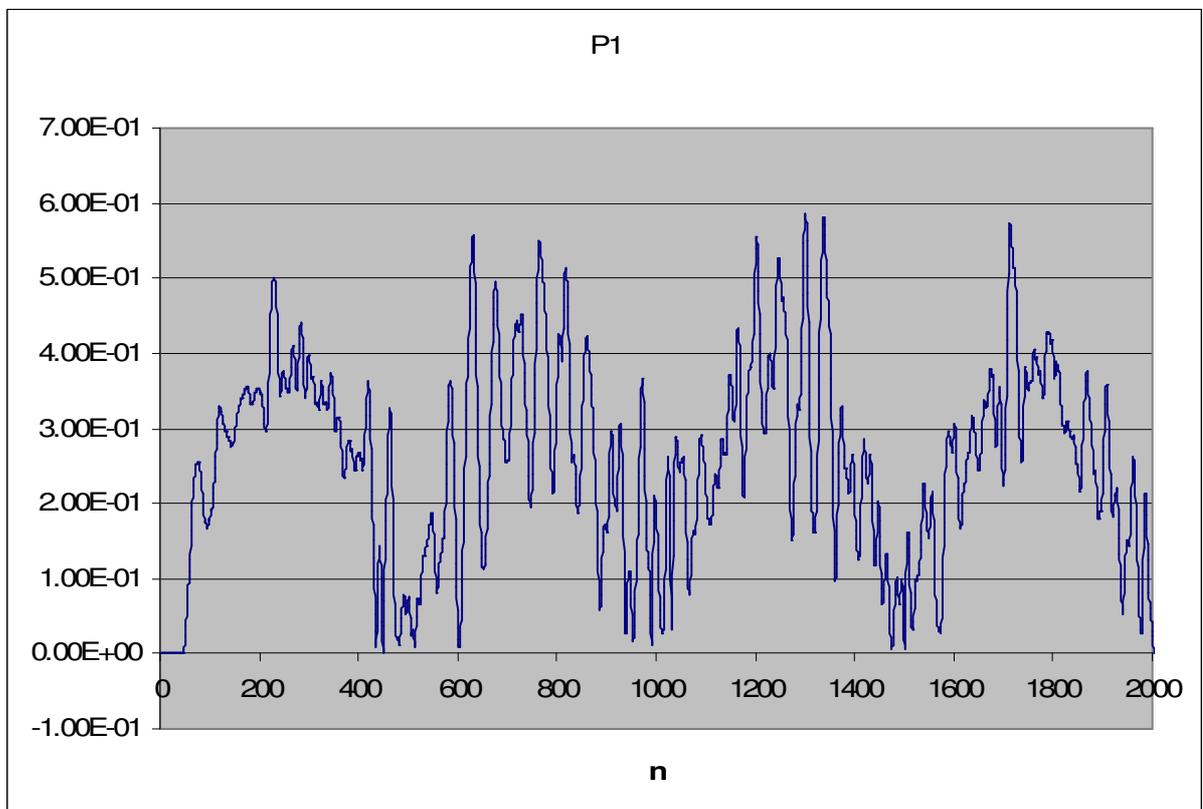


Fig. 19 – Simulação 2 ponto P1, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

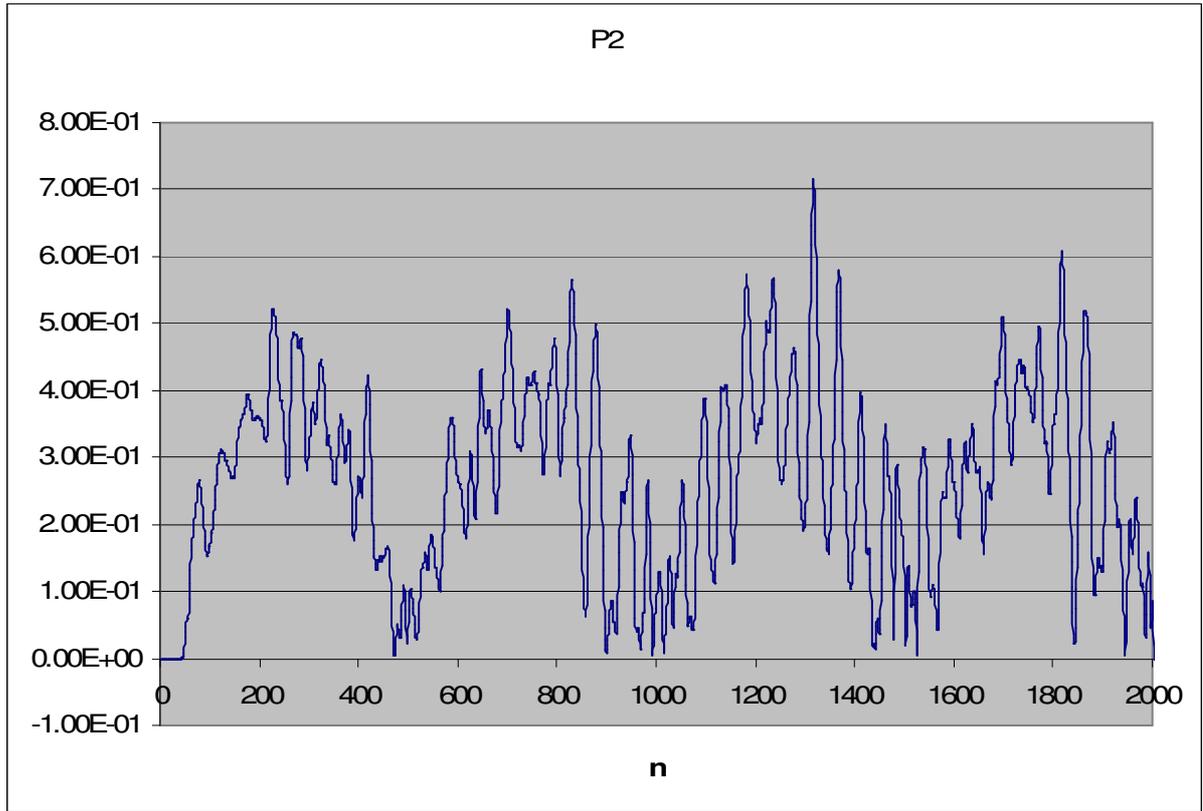


Fig. 20 – Simulação 2 ponto P2, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

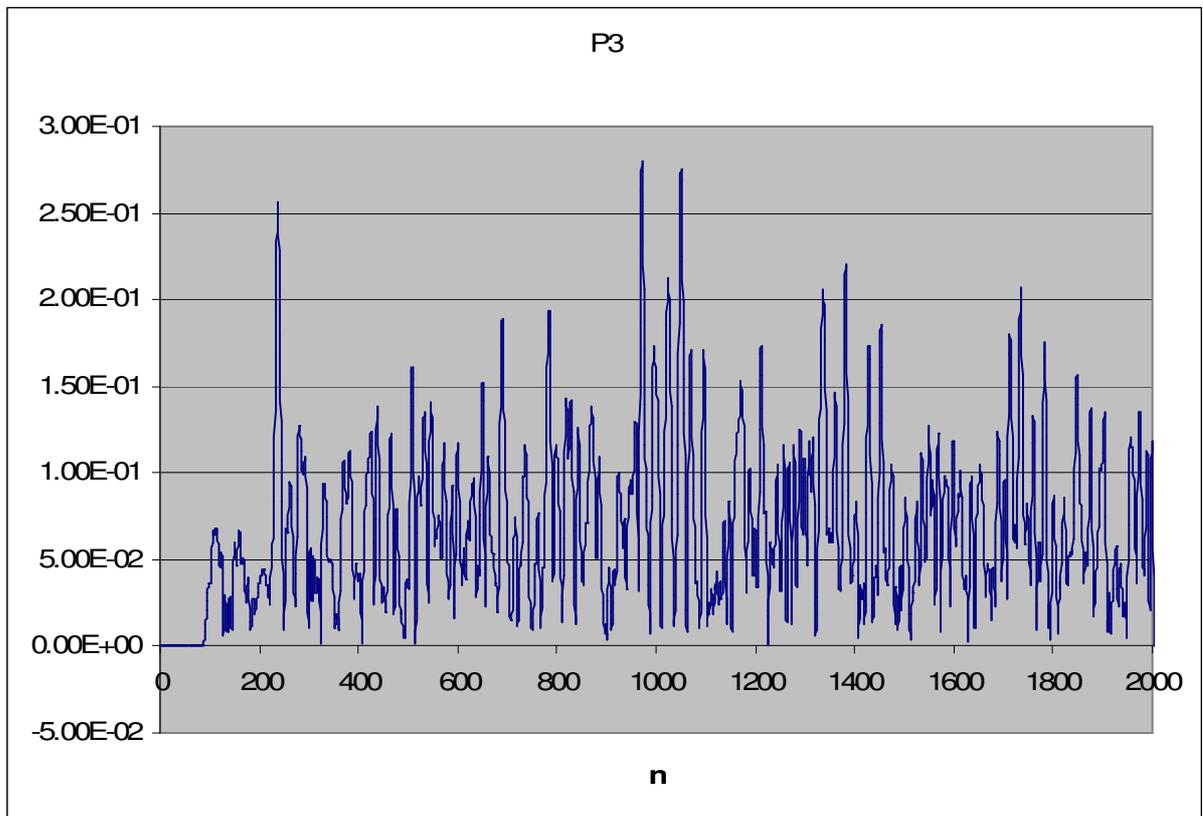


Fig. 21 – Simulação 2 ponto P3, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

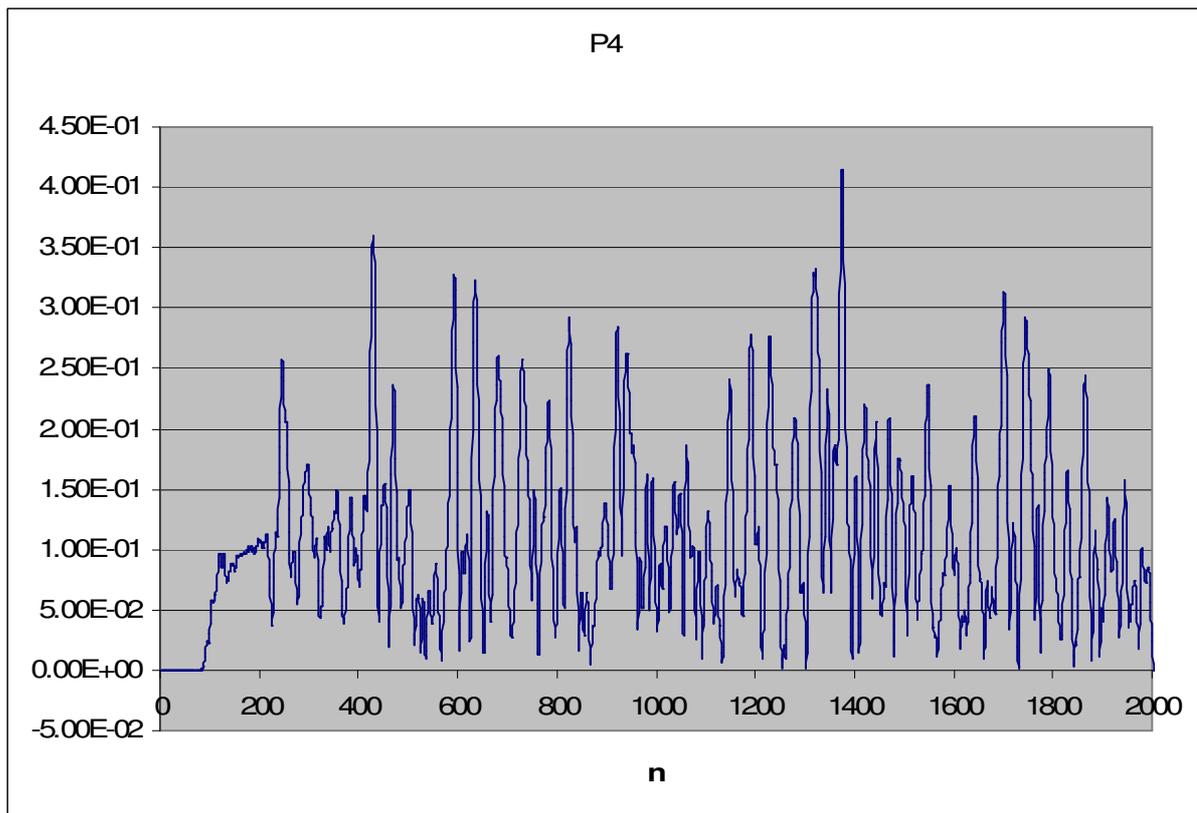


Fig. 22 – Simulação 2 ponto P4, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

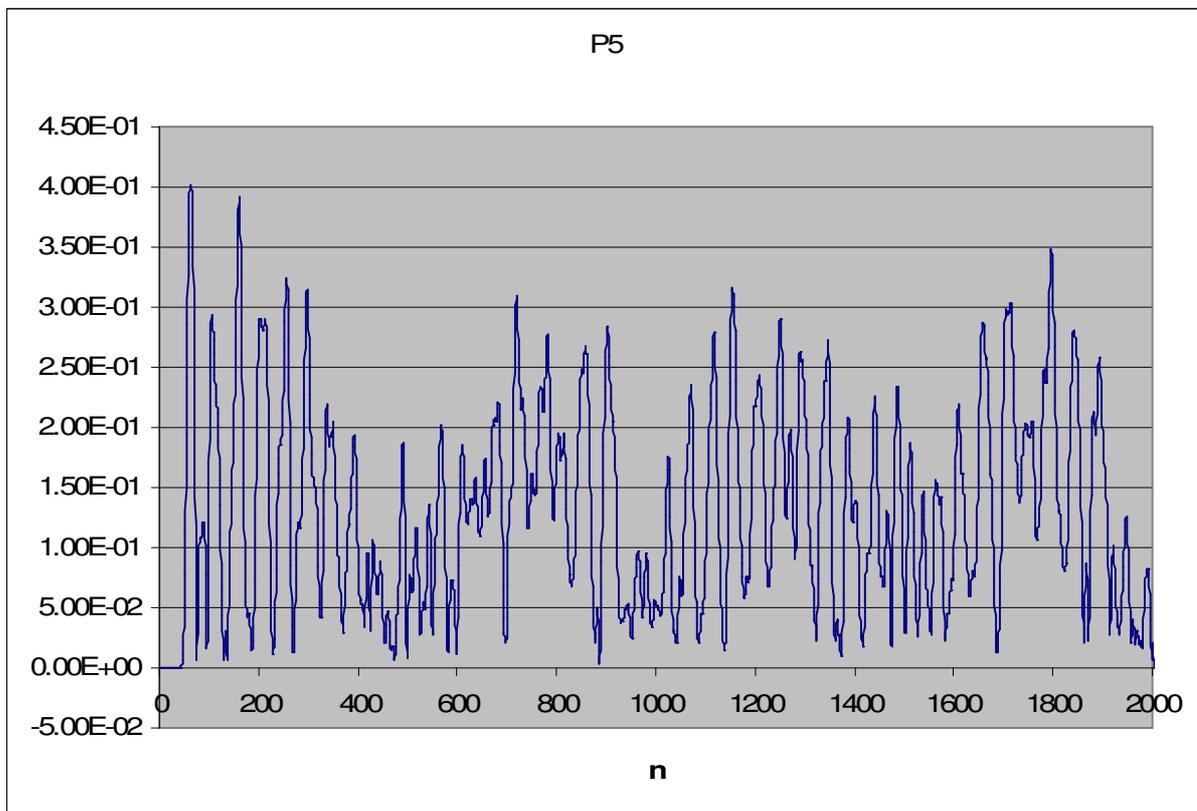


Fig. 23 – Simulação 2 ponto P5, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

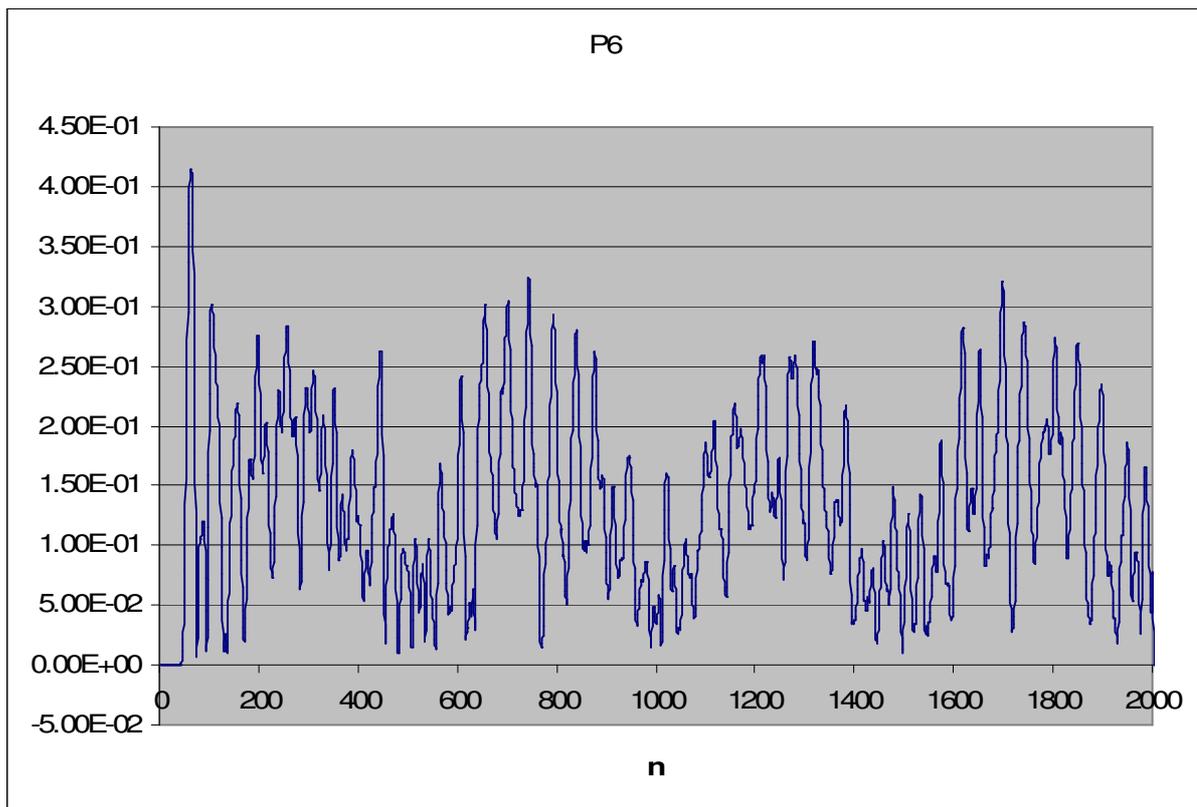


Fig. 24 – Simulação 2 ponto P6, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

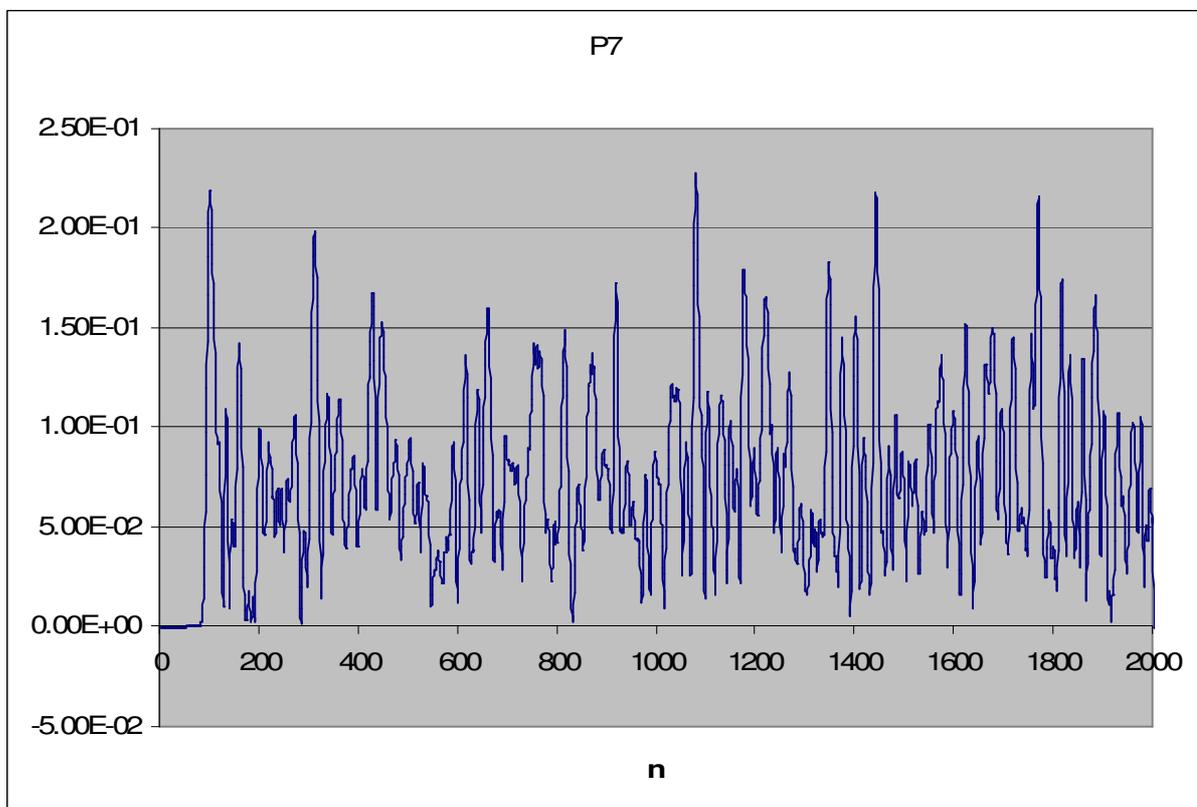


Fig. 25 – Simulação 2 ponto P7, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

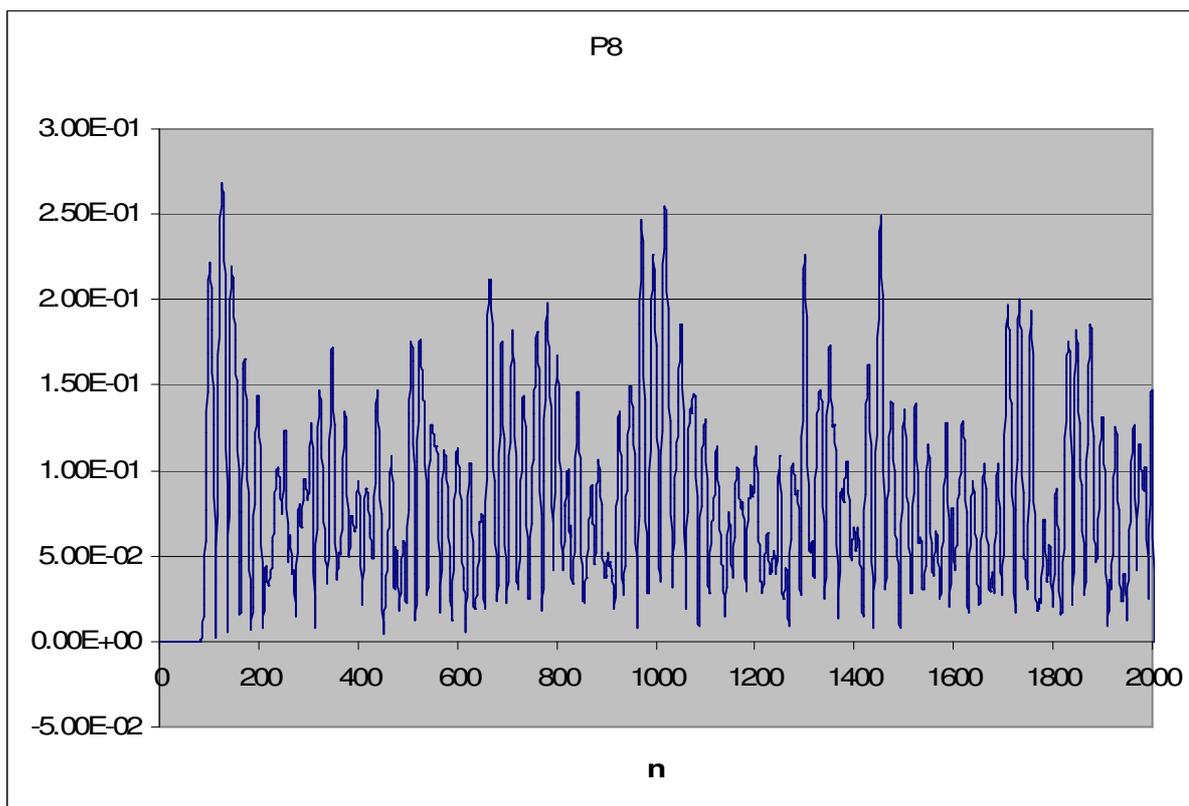


Fig. 26 – Simulação 2 ponto P8, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

Os resultados obtidos e distorções observadas nas duas simulações estão associados aos problema da reflexão numérica da onda nas bordas já referido, o que mostra a importância das técnicas de implementação da ABC (Absorbing Boundary Conditions) para permitir a utilização de *grids* compactos.

### b) Simulação do Algoritmo de Yee com UPML

O algoritmo Yee + *UPML* foi mais experimentado. Para não haver dúvidas com relação a possíveis erros numéricos advindos da programação, o primeiro teste realizado foi sem a presença de fontes ou corpos. Obviamente, os resultados esperados e encontrados foram valores nulos de campo em todo o *grid* durante todas as interações de tempo.

Na seqüência foram repetidos os ensaios da simulação de um dipolo curto no vácuo. Verificou-se que, após estabelecido o regime (após 300 interações) a onda atravessa o *grid*, com a forma e magnitude esperada.

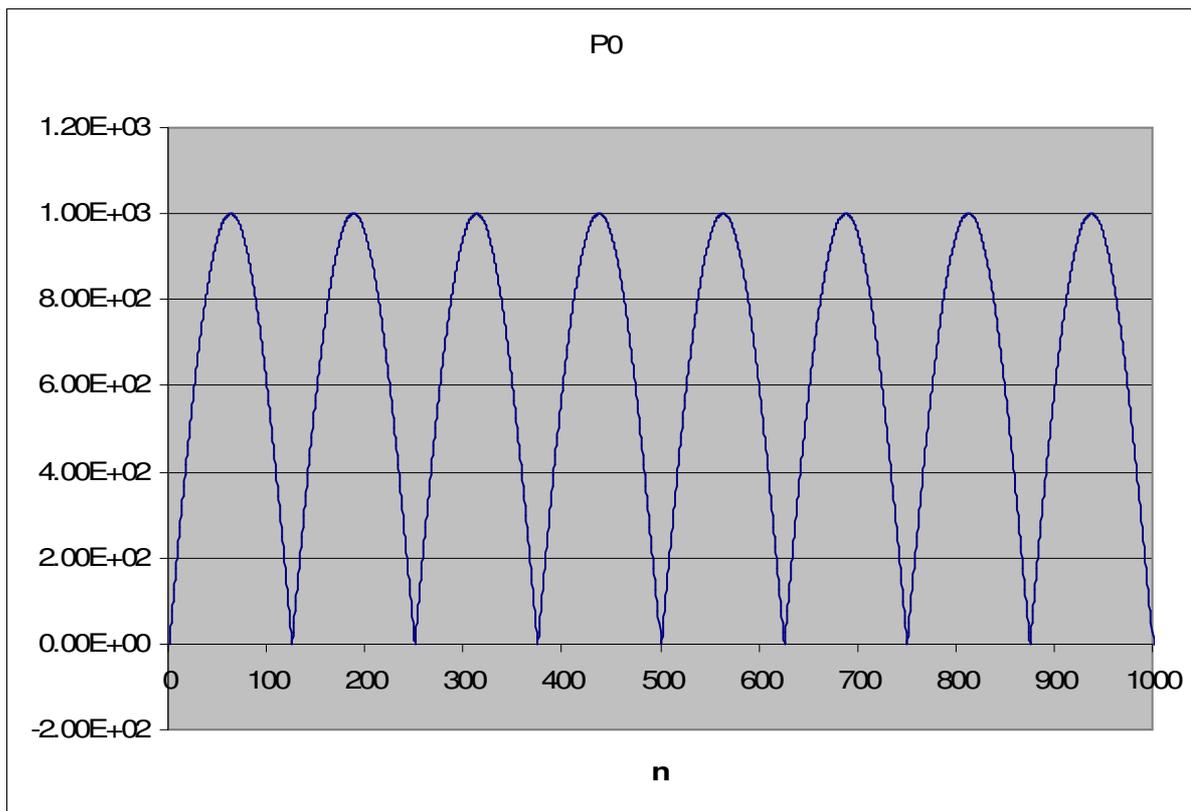


Fig. 27 – Simulação 3 ponto P0, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

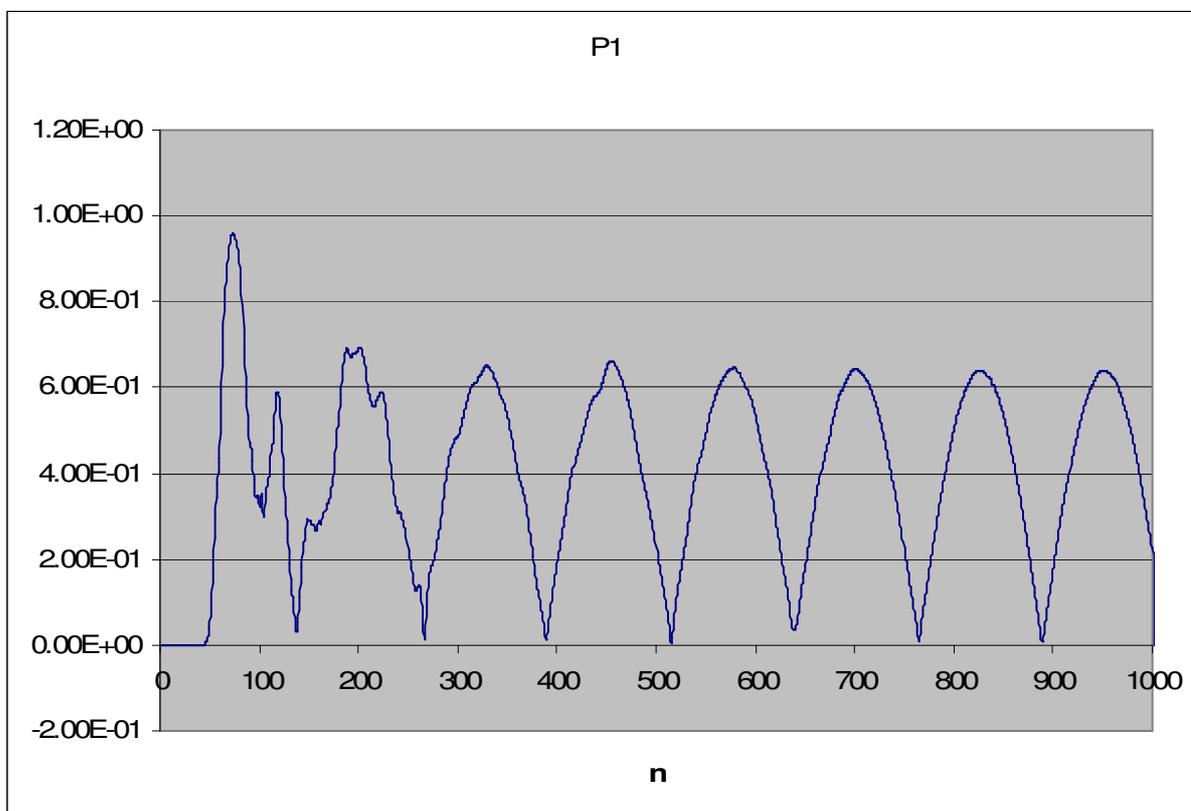


Fig. 28 – Simulação 3 ponto P1, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

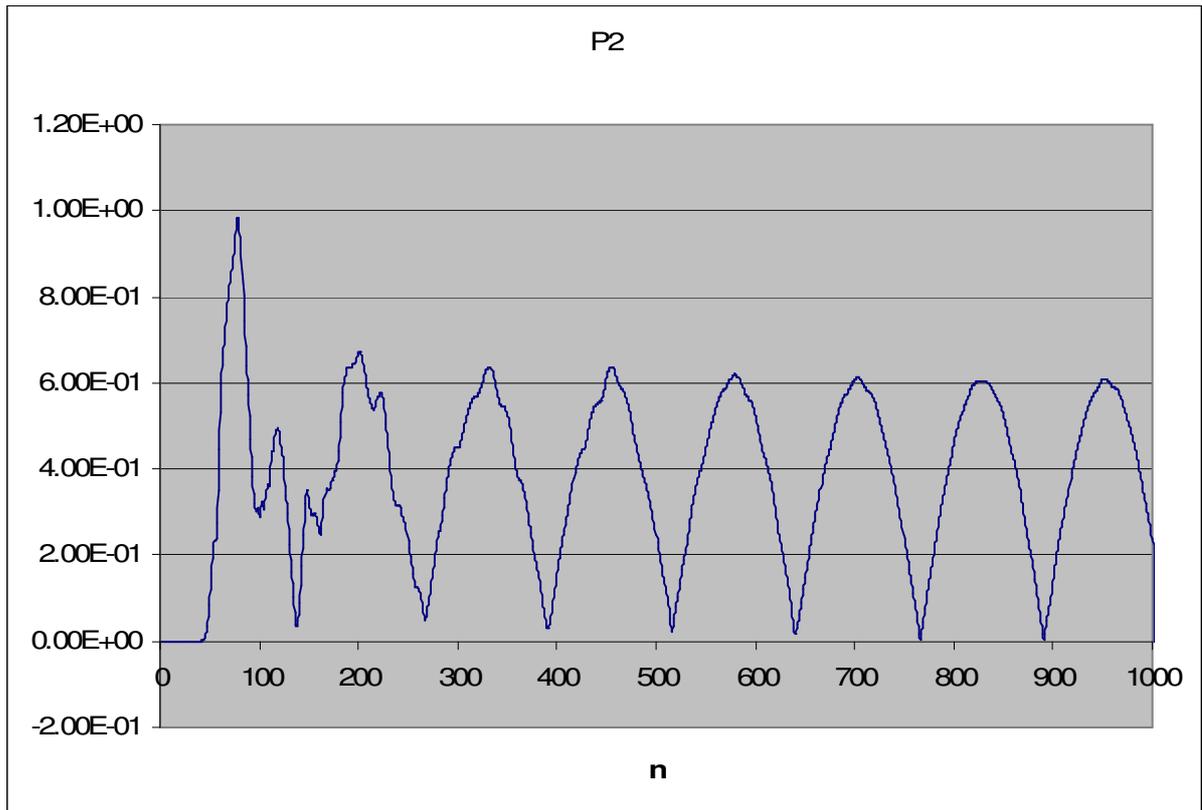


Fig. 29 – Simulação 3 ponto P2, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

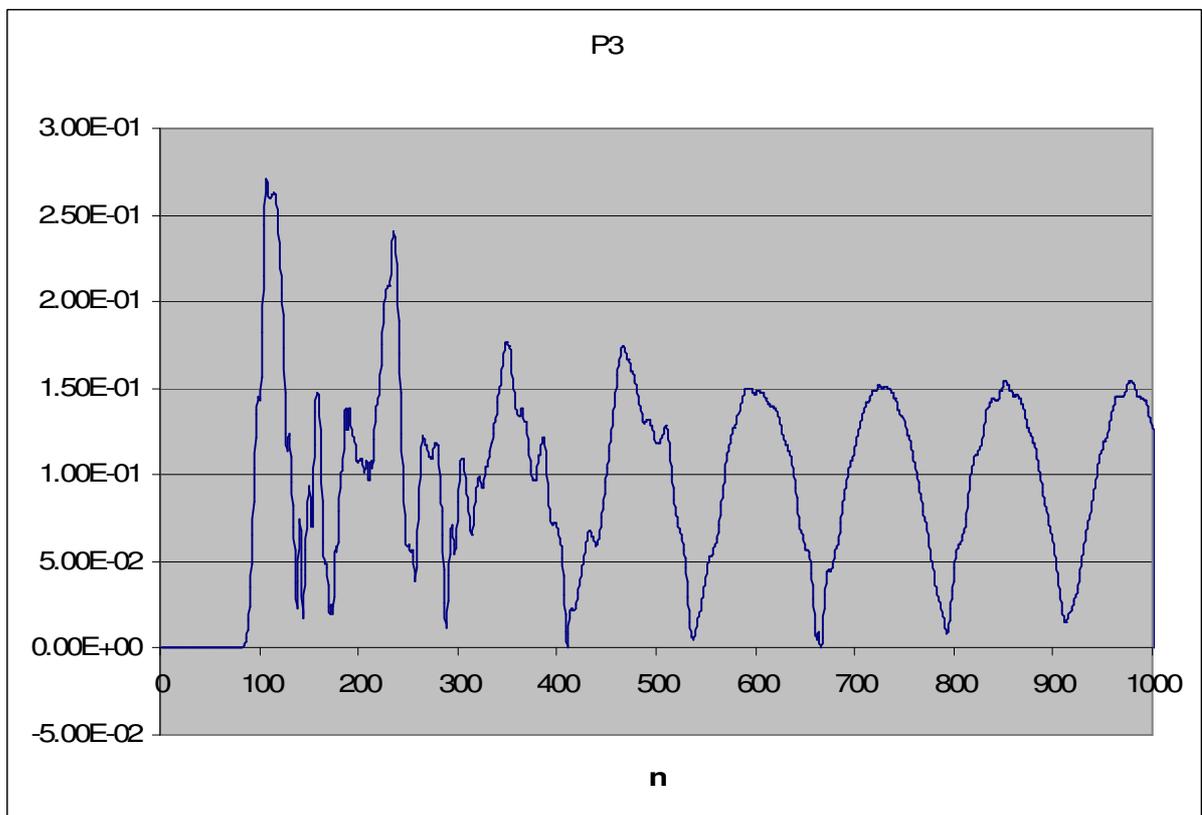


Fig. 30 – Simulação 3 ponto P3, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

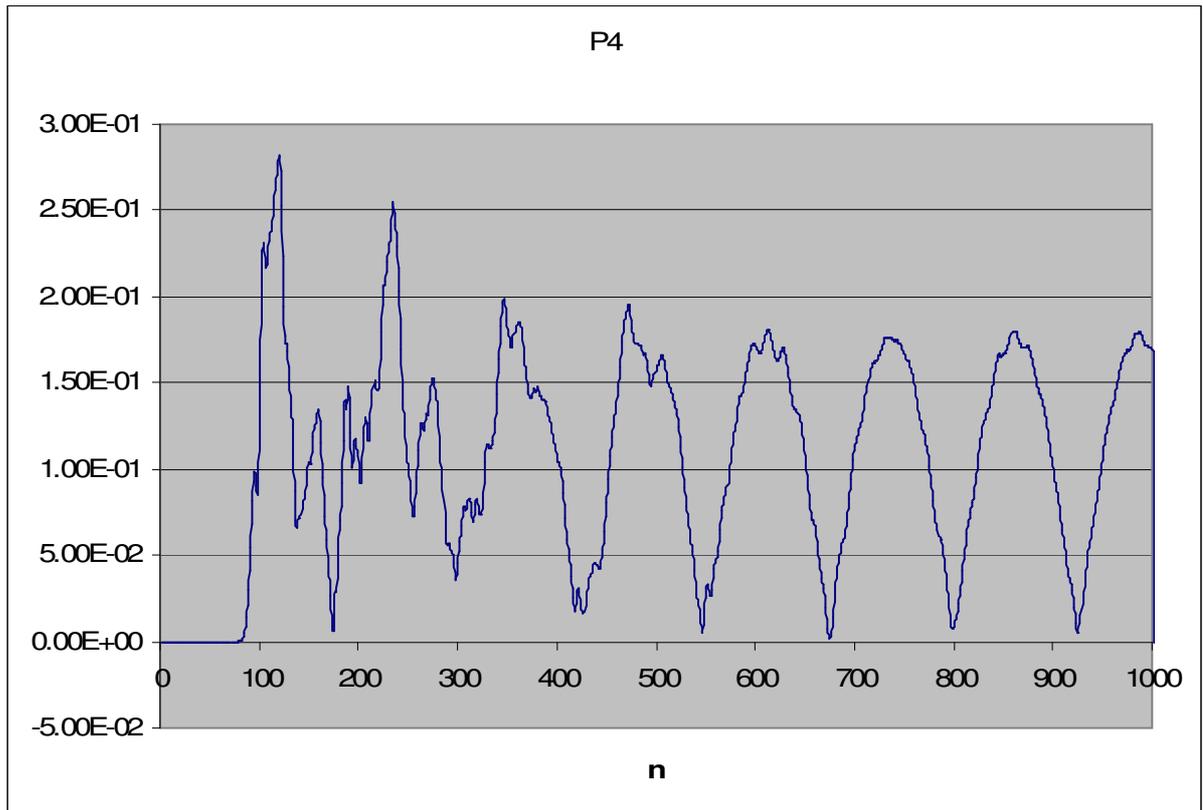


Fig. 31 – Simulação 3 ponto P4, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

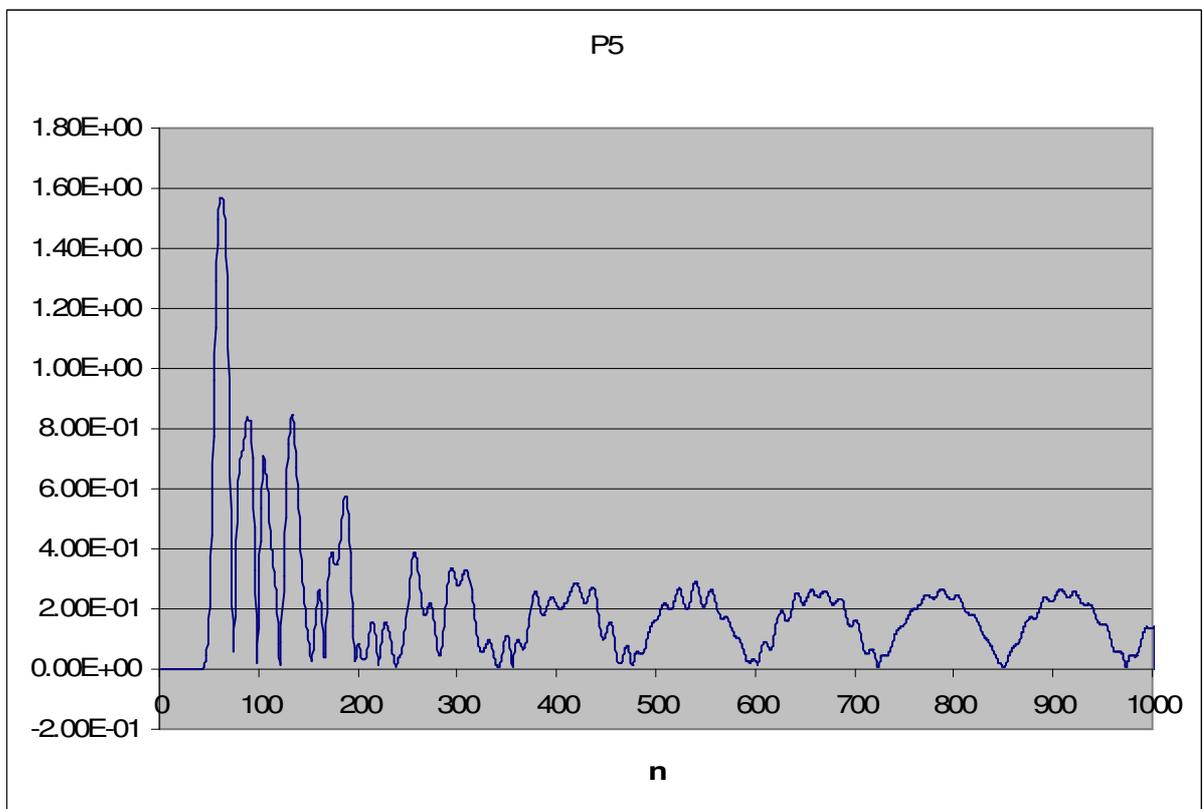


Fig. 32 – Simulação 3 ponto P5, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

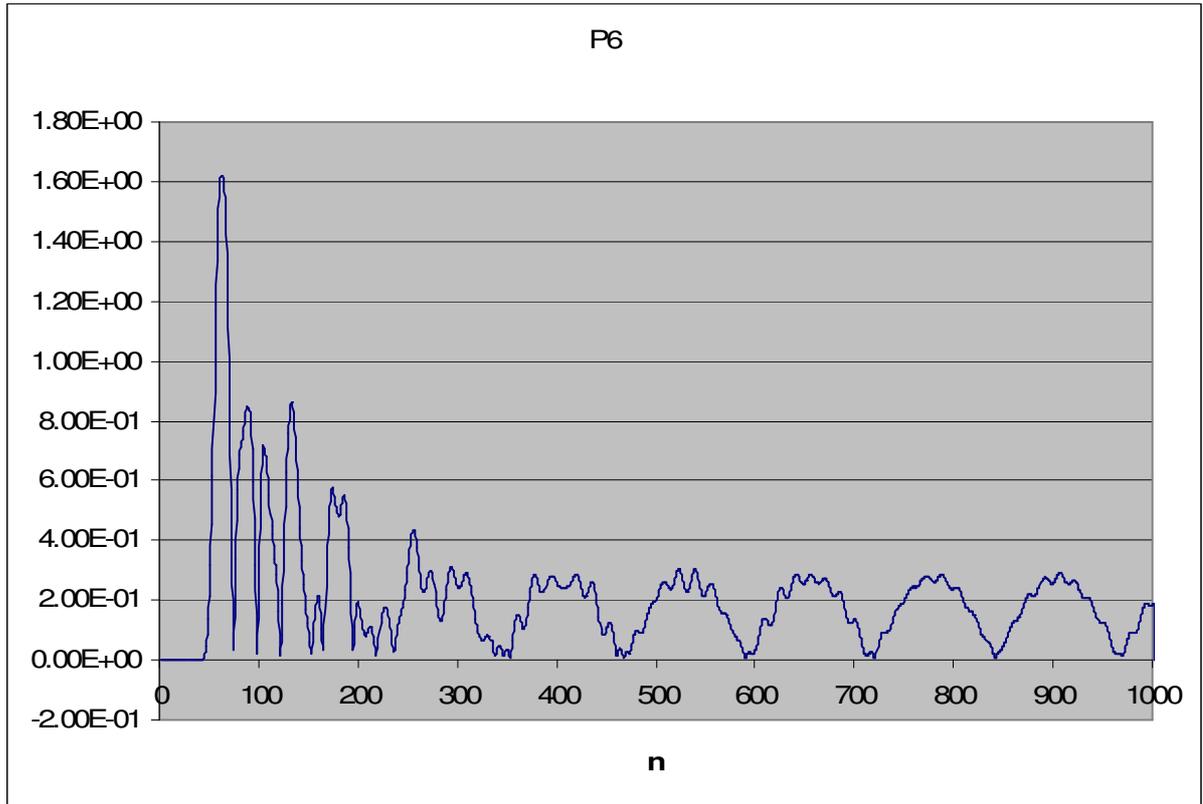


Fig. 33 – Simulação 3 ponto P6, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

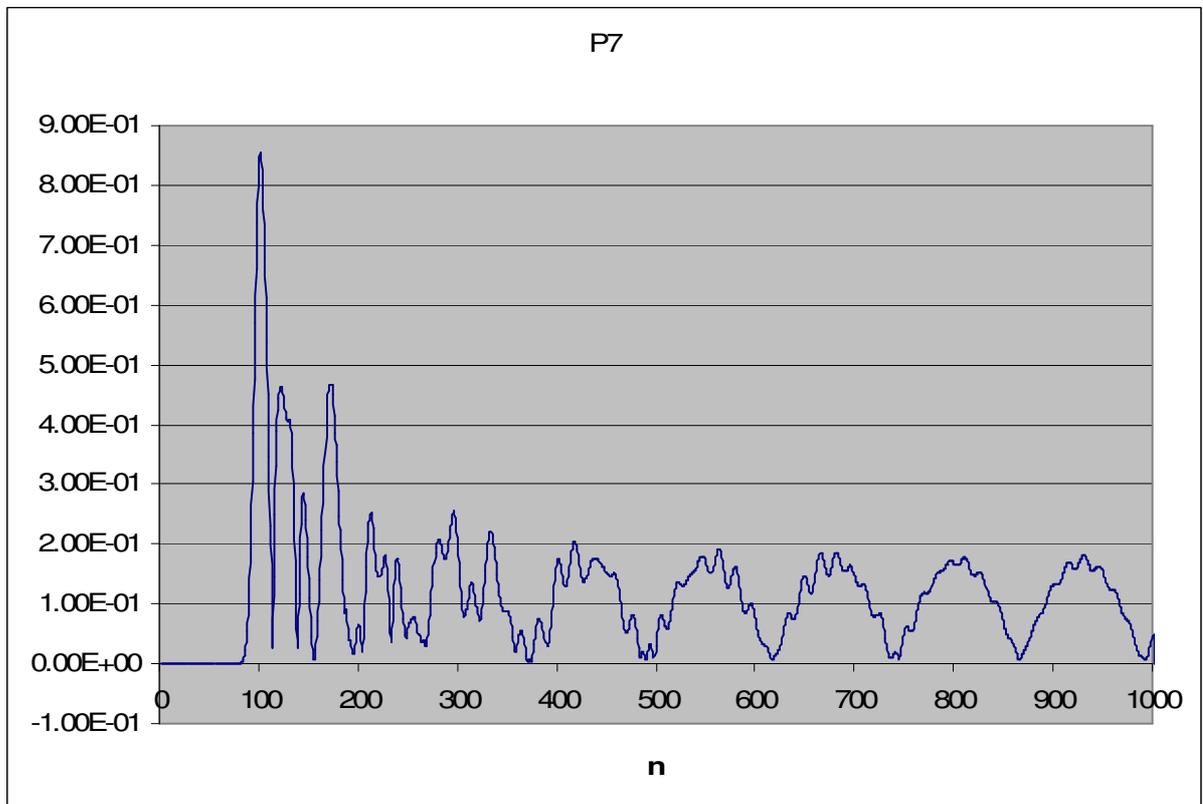


Fig. 34 – Simulação 3 ponto P7, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

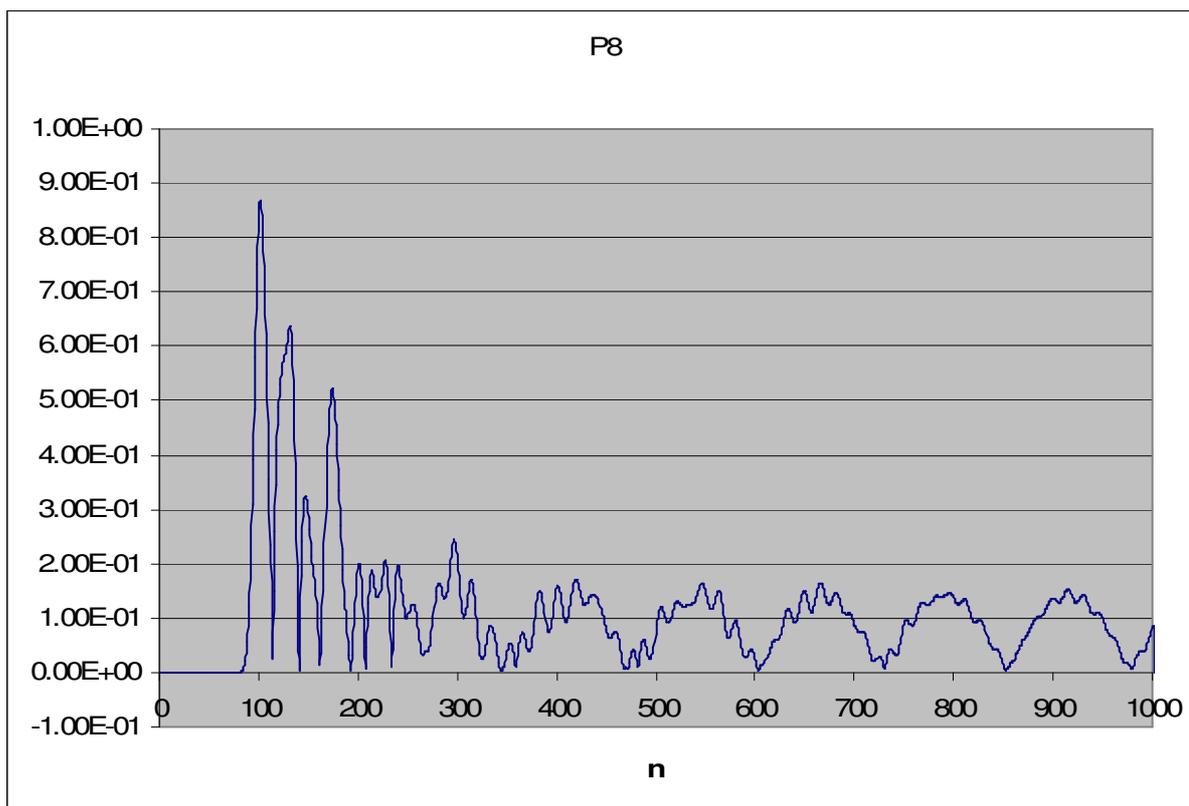


Fig. 35 – Simulação 3 ponto P8, campo elétrico resultante ( $\lambda=20\text{mm}$ ,  $\Delta=1\text{mm}$  e  $\Delta t=1,67\text{ps}$ )

Ainda notou-se a presença de ruído oscilatório, mas com magnitude muito menor ao existente nos resultados com o algoritmo sem *UPML*.

Ao aumentar-se a relação  $\lambda_{\text{wave}} / \lambda$  para 25, notou-se a presença de um ruído oscilatório, com magnitude na ordem de até  $2 \cdot 10^{-1}$  V/m, o qual desapareceu após 300 interações de tempo. O referido ruído era mais perceptível junto as bordas, provavelmente em função da dimensão dos valores de campo.

Ficou evidente que os resultados obtidos nessa última situação foram mais estáveis. Os valores típicos obtidos são mostrados a seguir

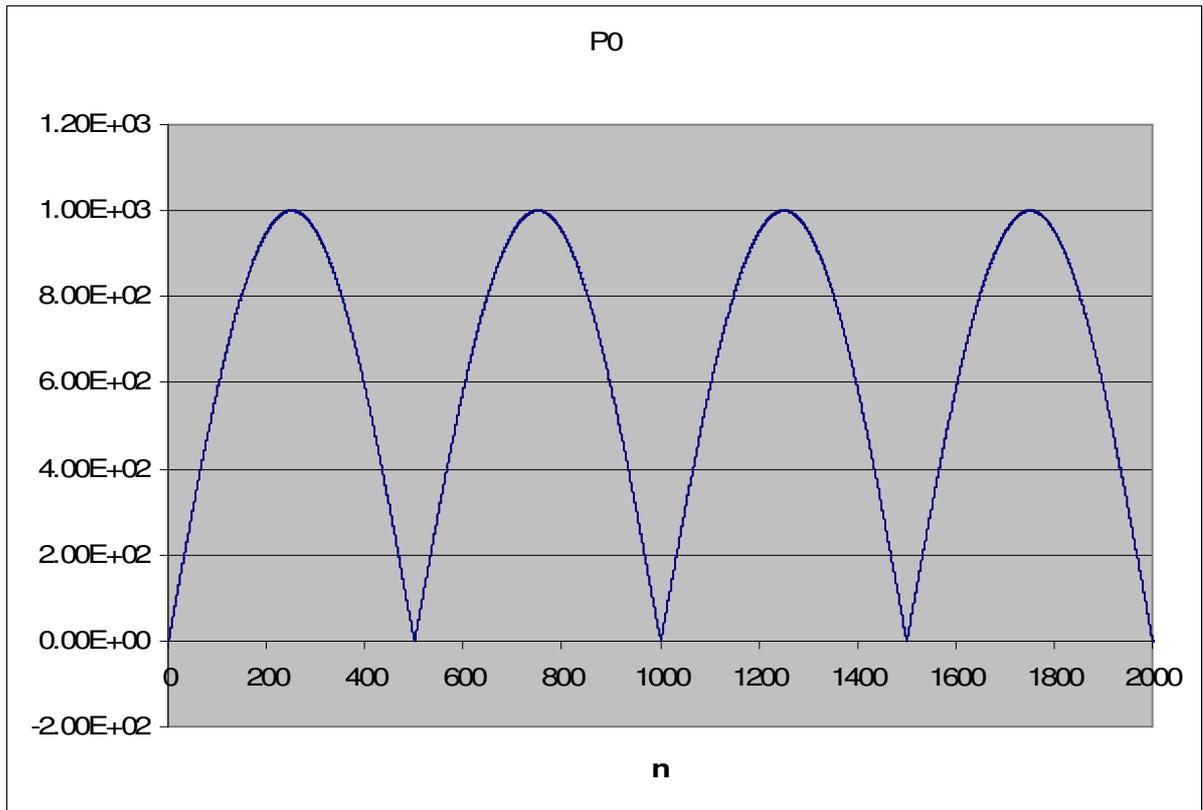


Fig. 36 – Simulação 4 ponto P0, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

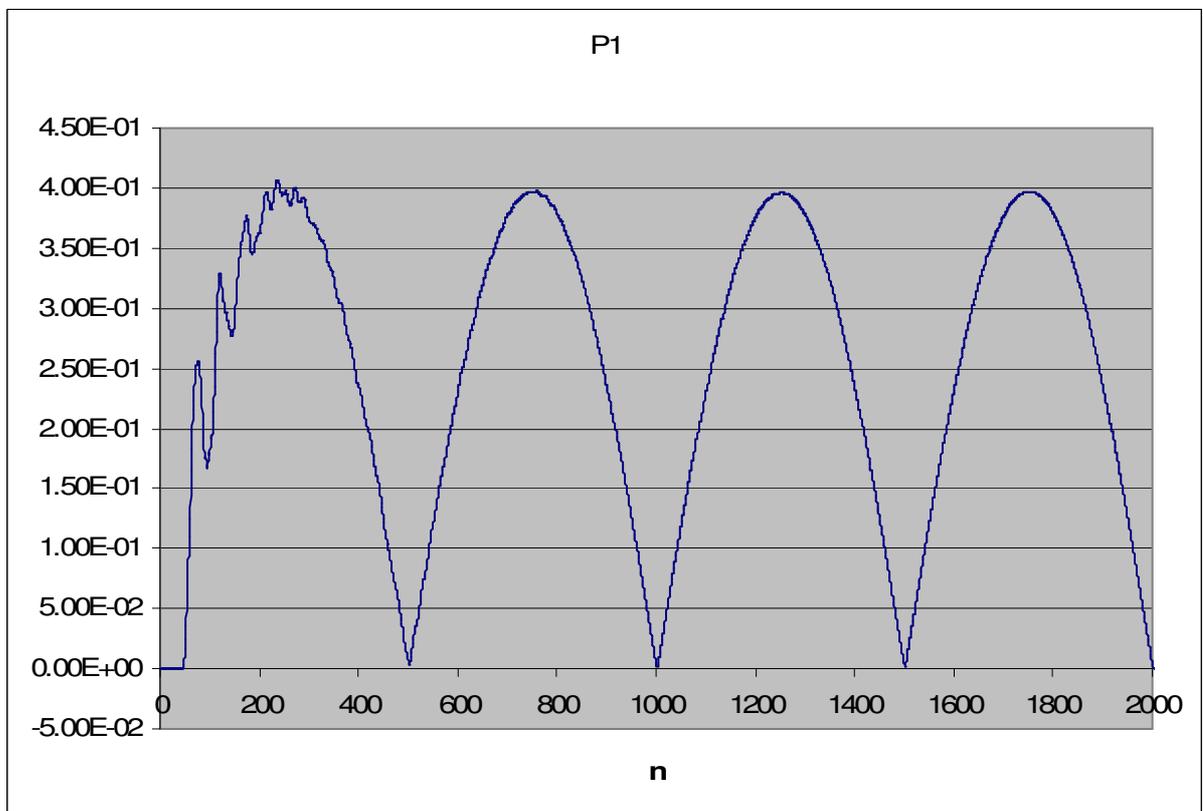


Fig. 37 – Simulação 4 ponto P1, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

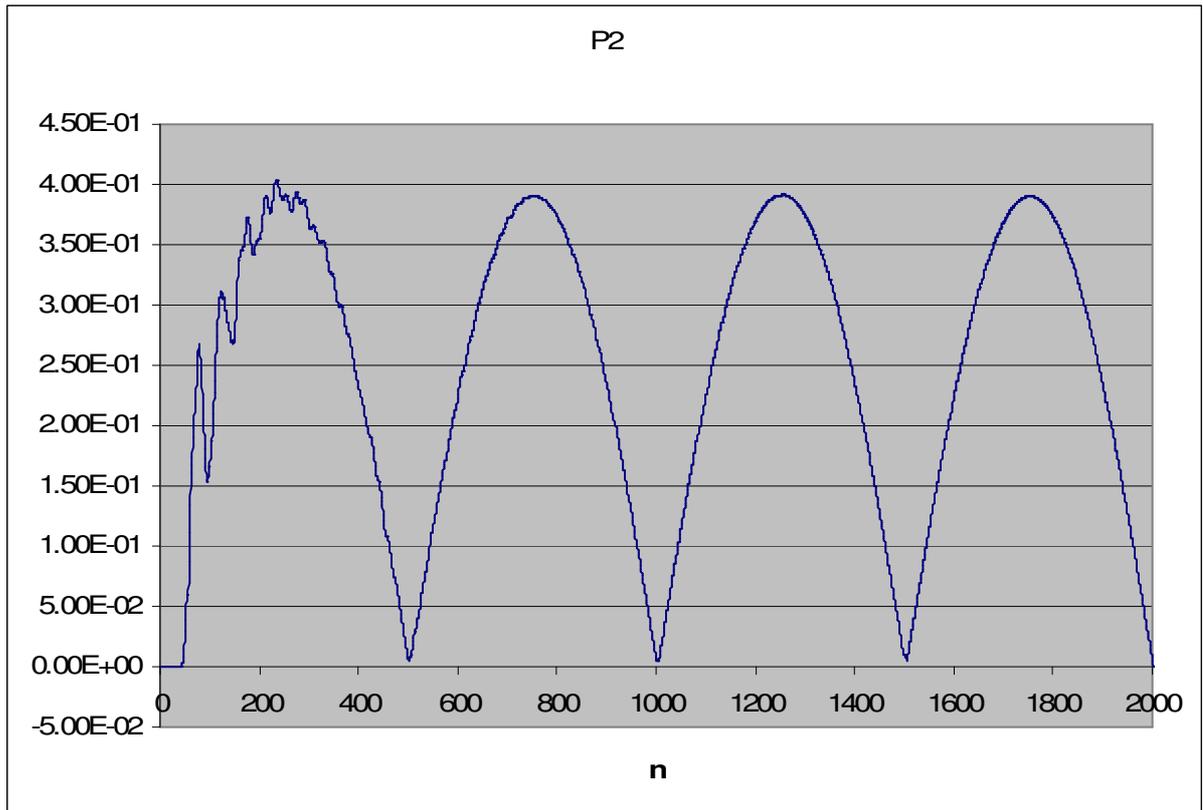


Fig. 38 – Simulação 4 ponto P2, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

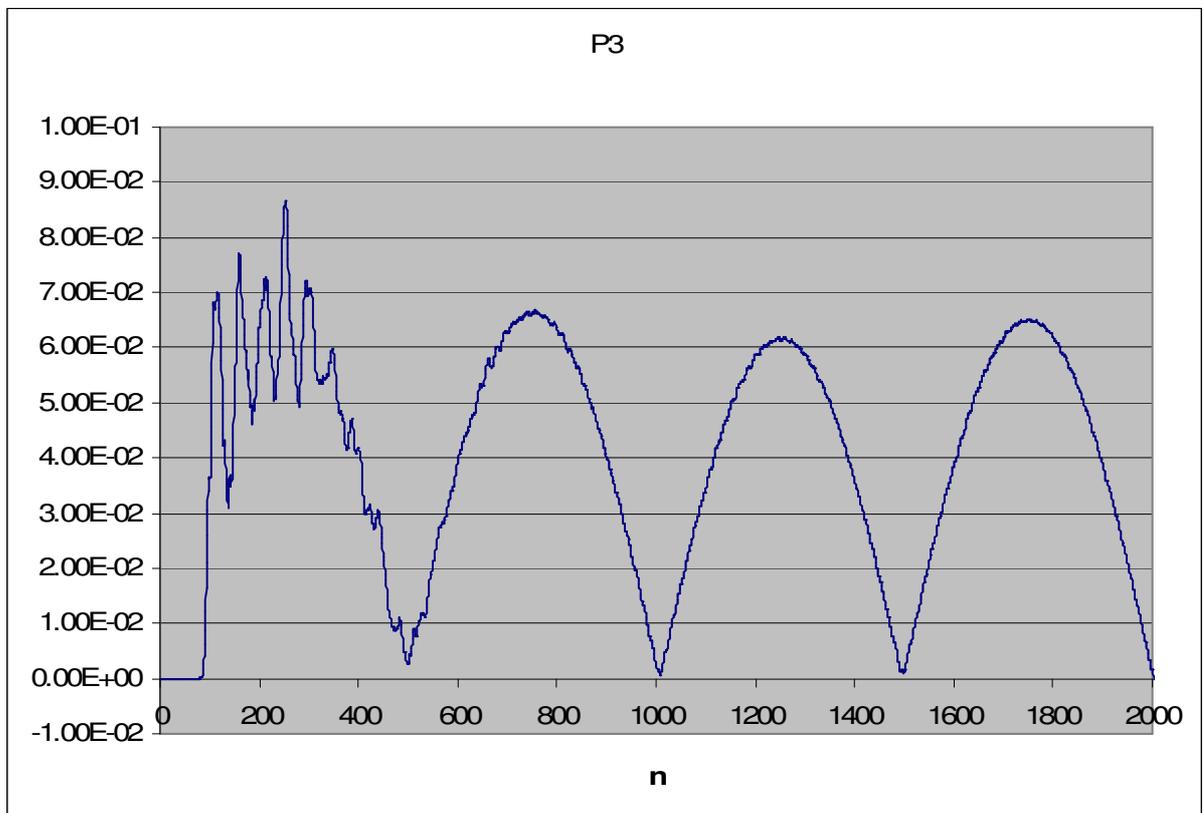


Fig. 39 – Simulação 4 ponto P3, intensidade do campo elétrico resultante

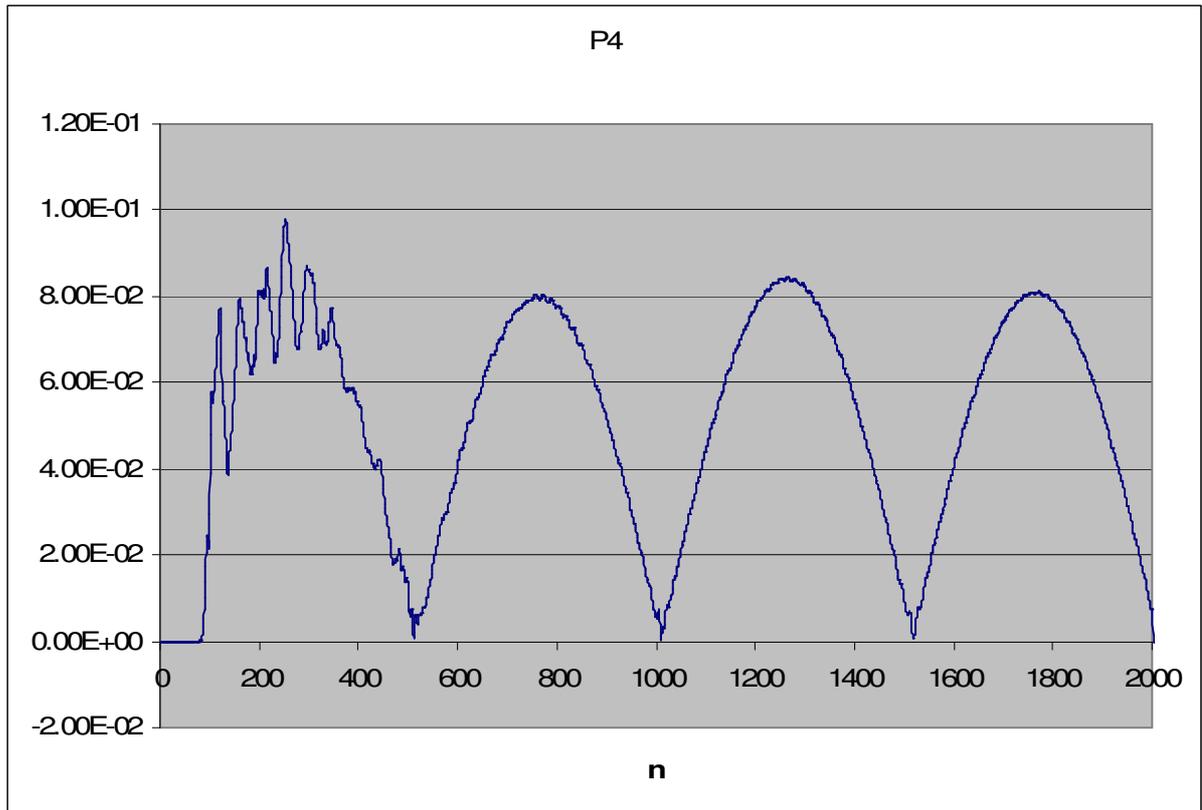


Fig. 40 – Simulação 4 ponto P4, campo elétrico resultante ( $\lambda=5mm$ ,  $\Delta=0,25mm$  e  $\Delta t=0,42ps$ )

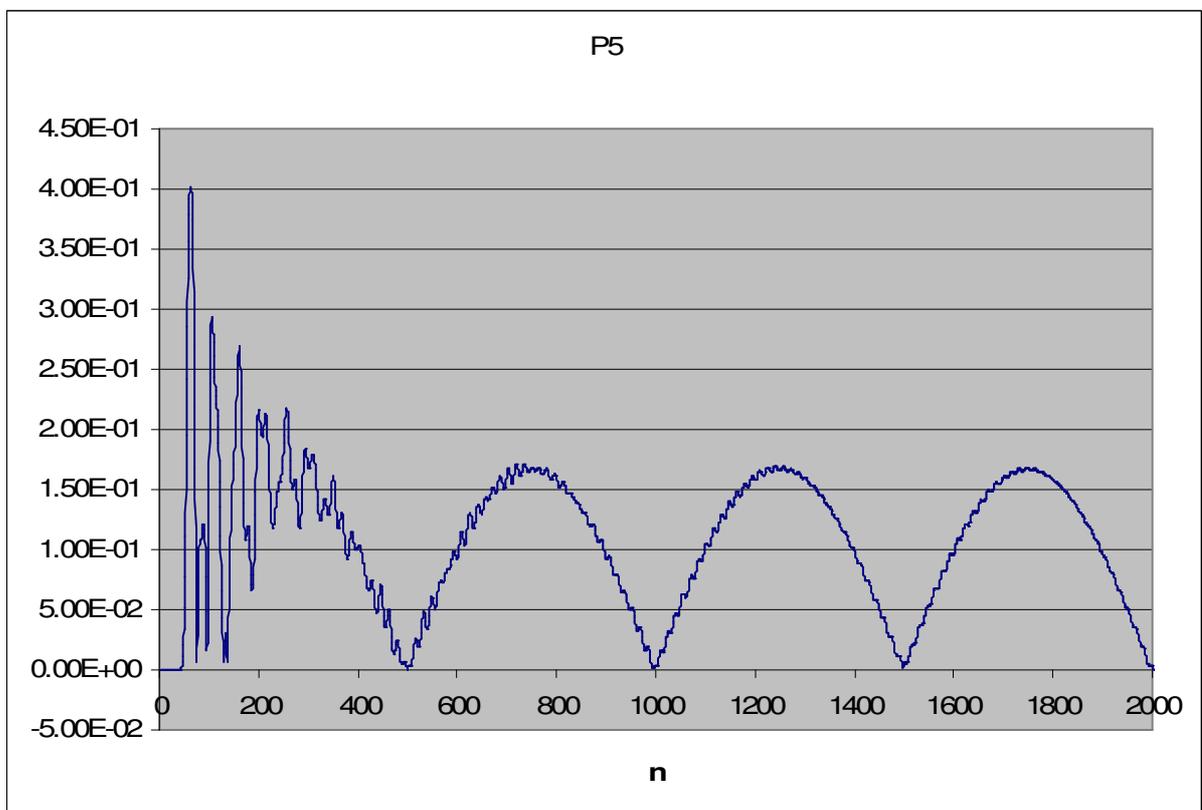


Fig. 41 – Simulação 4 ponto P5, campo elétrico resultante ( $\lambda=5mm$ ,  $\Delta=0,25mm$  e  $\Delta t=0,42ps$ )

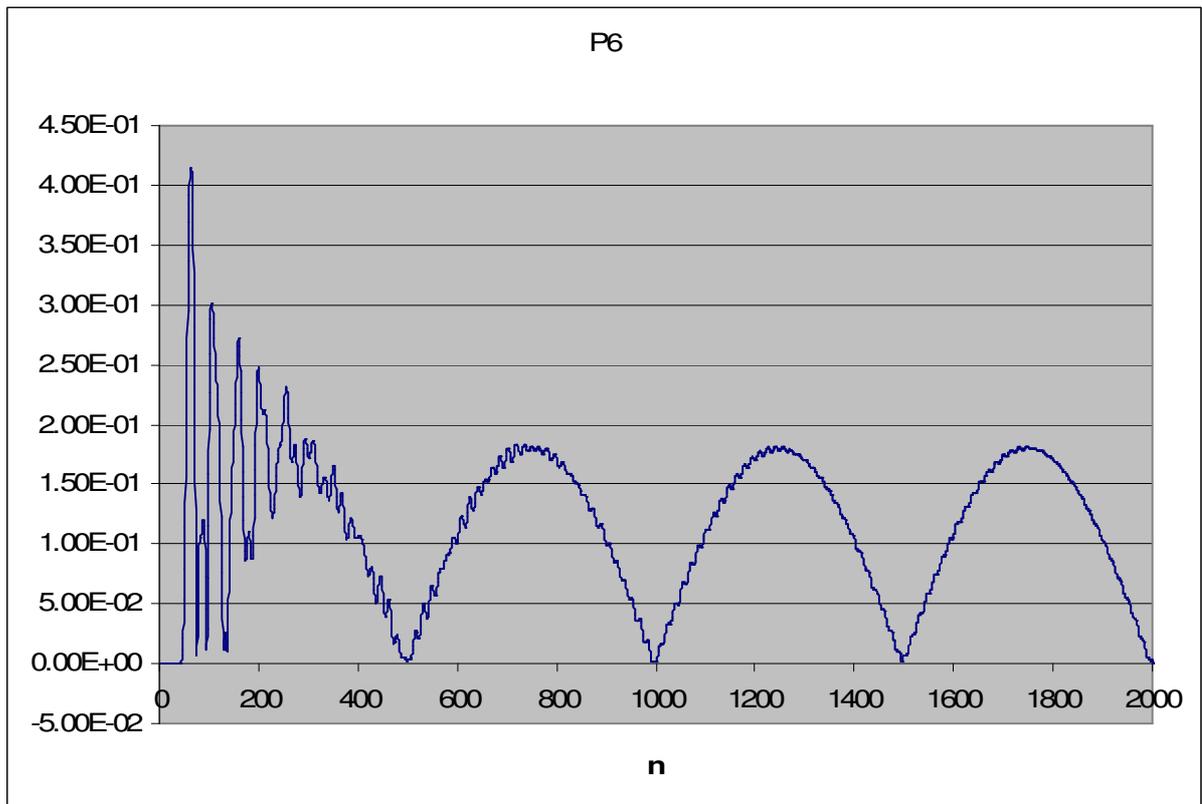


Fig. 42 – Simulação 4 ponto P6, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

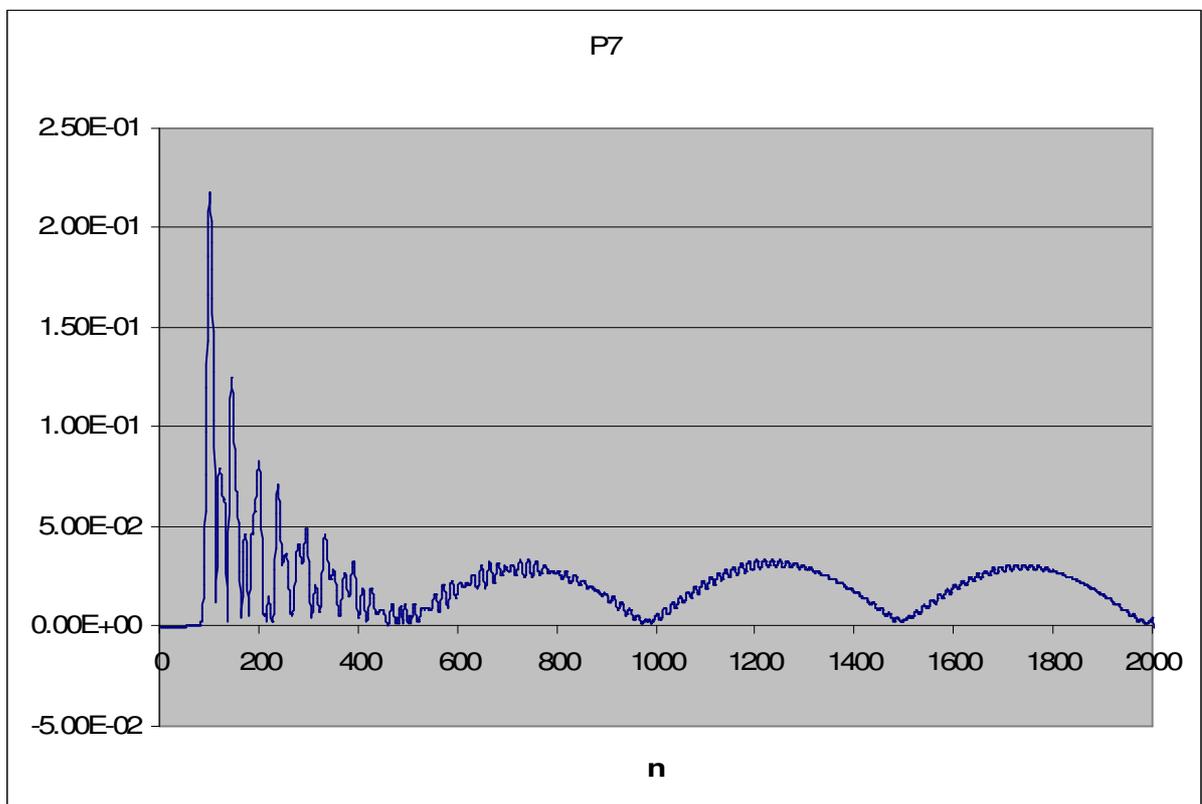


Fig. 43 – Simulação 4 ponto P7, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

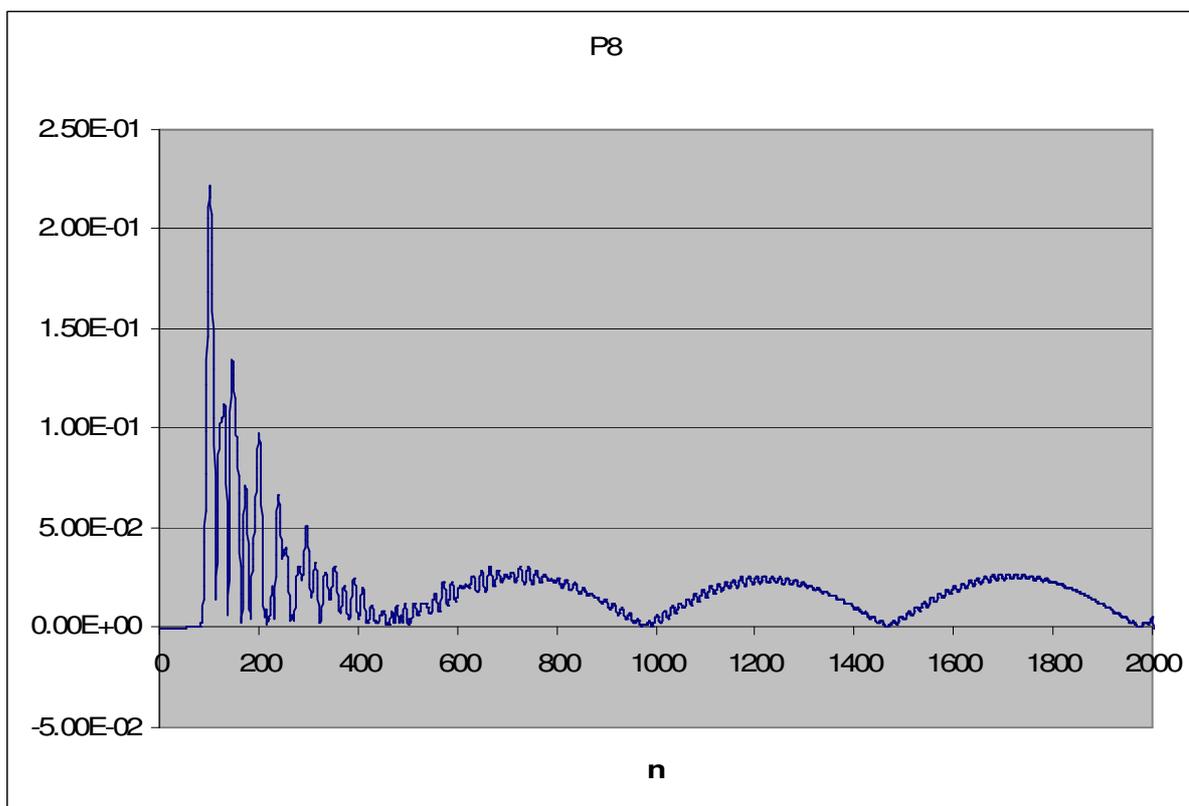


Fig. 44 – Simulação 4 ponto P8, campo elétrico resultante ( $\lambda=5\text{mm}$ ,  $\Delta=0,25\text{mm}$  e  $\Delta t=0,42\text{ps}$ )

Foi investigada ainda a influência no número de camadas absorventes sobre o resultado, na expectativa de que os resultados obtidos para uma *UPML* com mais células pudessem ser melhores que aqueles para uma *UPML* mais fina, por haver uma diferença menor do valor de condutividade entre as camadas contíguas

Contudo, nos ensaios realizados na simulação dos sistemas descritos, para uma camada absorvente de 10 células ( $N_{upml} = 10$ ), não foi evidenciada diferença significativa. Os resultados foram praticamente iguais aos obtidos para  $N_{upml} = 5$ .

Outro teste realizado foi o estudo da propagação em *grids* formado por materiais diferentes do vácuo. Quando o material era o ar, cujas características são próximas a do vácuo, os resultados foram muito próximos aos valores obtidos nos ensaios de propagação no vácuo.

Quando o material era um condutor (por exemplo, um cubo de cobre), não houve a propagação da onda no interior do *grid*, como era de se esperar. Abaixo é

apresentada uma lista de materiais e suas propriedades eletromagnética (para  $f < 10$  GHz e  $T=25^{\circ}\text{C}$ ):

Tab. 16 – Propriedades eletromagnéticas de diversos materiais

$\sigma$	$\epsilon_r$	$\mu_r$	$\sigma^*$	Descrição
0.0E+0	1.0	1.0	0.0E+0	vácuo
1.0E-30	1.000054	1.0000004	0.0E+0	ar (ridigez = 0.8 kV/mm)
6.70E+7	1.0	0.99998	0.0E+0	prata
5.85E+7	1.0	0.999991	0.0E+0	cobre
4.20E+7	1.0	1.0	0.0E+0	ouro
3.50E+7	1.0	1.00002	0.0E+0	alumínio
1.37E+7	1.0	600.0	0.0E+0	niquel
1.02E+7	1.0	5000.0	0.0E+0	ferro
0.95E+7	1.0	1.0	0.0E+0	platina
6.67E+4	1.0	1.0	0.0E+0	carbono
4.00E+0	81.0	1.0	0.0E+0	agua do mar
2.20E+0	16.0	1.0	0.0E+0	germanio (puro)
1.00E-3	12.0	1.0	0.0E+0	silício (puro)
1.00E-3	10.0	1.0	0.0E+0	terra úmida (apenas referência, sem material ferromagnético, pois depende composição)
1.00E-5	5.0	1.0	0.0E+0	terra seca (apenas referência, sem material ferromagnético, pois depende composição)
1.00E-6	81.0	1.0	0.0E+0	agua destilada
1.0E-16	4.0	1.0	0.0E+0	vidro Corning 707 (ridigez = 13 kV/mm - Pirex)
1.0E-16	3.78	1.0	0.0E+0	quartzo fundido (ridigez = 8 kV/mm)
1.0E-16	5.4	1.0	0.0E+0	mica ruby (ridigez = 160 kV/mm)
1.0E-16	4.95	1.0	0.0E+0	cerâmica Alsimag 393
1.0E-16	100	1.0	0.0E+0	dióxido de titânio (ridigez = 6 kV/mm)
1.0E-16	2.56	1.0	0.0E+0	polistireno (ridigez = 25 kV/mm)
1.0E-16	5.7	1.0	0.0E+0	neoprene (ridigez = 12 kV/mm)
1.0E-15	2.3	1.0	0.0E+0	polietileno (ridigez = 50 kV/mm)
1.0E-16	2.1	1.0	0.0E+0	(ridigez = 60 kV/mm)
1.0E-11	4.8	1.0	0.0E+0	baquelite (ridigez = 12 kV/mm)
1.0E-16	6.5	1.0	0.0E+0	porcelana (ridigez = 4 kV/mm)
1.0E-16	3.5	1.0	0.0E+0	papel (ridigez = 14 kV/mm)

Por último, para fins de visualização do campo irradiado na região em torno de uma antena, foi realizada a comparação dos resultados de simulação obtidos os valores previstos pelo modelo teórico para dipolos curtos.

Os resultado de simulação obtidos foram comparados com a expressão teórica para cálculo do campo elétrico no entorno de um dipolo curto sob excitação senoidal, considerando campo distante ( $r > \lambda_{\text{wave}} / 2\pi$ ), as quais estão apresentadas abaixo:

$$E_{\theta} = \frac{I_0 \ell e^{j(\omega t - \beta r)} \sin \theta}{4\pi \epsilon_0} \left( \frac{j\omega}{c^2 r} \right) = \frac{I_0 \ell e^{j\left(\omega t - \beta r + \frac{\pi}{2}\right)} \sin \theta}{2\epsilon_0 c \lambda r} \left[ \frac{\text{V}}{\text{m}} \right] \quad (\text{Eq. 135})$$

$$E_r = 0 \quad (\text{Eq. 136})$$

$$E_\phi = 0 \quad (\text{Eq. 137})$$

As expressões acima referidas tem sua demonstração detalhada em [12] e pressupõe a representação do sistema em coordenadas esféricas, conforme ilustrado na figura a seguir.

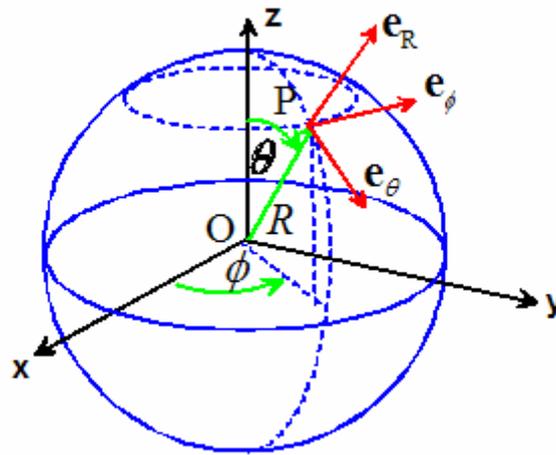


Fig. 45 – Sistema de coordenadas esféricas

A necessária transposição das componentes em coordenadas cartesianas para o sistema de coordenadas esféricas pode ser realizada a partir da seguinte relação:

$$\begin{bmatrix} E_r \\ E_\theta \\ E_\phi \end{bmatrix} = \begin{bmatrix} \sin \theta \cos \phi & \sin \theta \sin \phi & \cos \theta \\ \cos \theta \cos \phi & \cos \theta \sin \phi & -\sin \theta \\ -\sin \phi & \cos \phi & 0 \end{bmatrix} \begin{bmatrix} E_x \\ E_y \\ E_z \end{bmatrix} \quad (\text{Eq.136})$$

Para tentar alcançar a condição de campo distante, foi adotado um grid de 20x200x200,  $\lambda=50\text{mm}$ ,  $\Delta=2,5\text{mm}$  e  $\Delta t=4,17\text{ps}$ .

A localização e geometria do dipolo foi mantida, conforme anteriormente apresentado.

Considerou-se os pontos de amostragem de interesse P5 a P8, onde tem-se  $\theta=90^\circ$ ,  $\phi=90^\circ$  ou  $\phi=270^\circ$ . Com efeito, tem-se dessa forma  $|E_z| = |E_\theta|$ .

Além disso, na condição de campo distante, a seguinte relação deve ser preservada:

$$\frac{E_{\theta}(r_1)}{E_{\theta}(r_2)} = \frac{r_2}{r_1} \quad (\text{Eq. 139})$$

Ainda, por aproximação, foi adotado:

$$I_o = \frac{E_o \cdot \Delta}{Z_o} \quad (\text{Eq.140})$$

onde:

$Z_o = 120\pi$  : impedância do dipolo curto.

Dessa forma,  $I_o = 0,00663\text{A}$  e  $E_{\theta} = 0,1748 / r$ . Os resultados teóricos e os obtidos em simulação estão apresentados nas tabelas abaixo.

Tab. 17 – Comparação entre os valores de campo teóricos e resultados obtidos

Distância	Teórico	Simulação
$E_{\theta} (50\Delta = 125 \text{ mm})$	1,398 V/m	1,32 V/m
$E_{\theta} (95\Delta = 237,5 \text{ mm})$	0,736 V/m	0,70 V/m

Tab. 18 – Variação dos valores de campo com a distância

	Teórico	Simulação
$E_{\theta} (r_1) / E_{\theta} (r_2)$	1,900	1,886

Percebe-se que, apesar dos erros em torno de 5% verificados, a simulação apresenta resultados próximos aos valores teóricos esperados. Esses erros são justificáveis pela série de aproximações realizadas.

Porém, um aspecto mais importante que os valores absolutos de campo foi confirmado com menor erro. Verificou-se que a variação do campo em função da distância está coerente com o resultado teórico, apresentando uma diferença inferior a 0,8%.

## 8. Conclusão

O resultado final do trabalho é um software genérico, que realiza a simulação numérica da interação entre ondas eletromagnéticas e corpos em quatro dimensões (três espaciais e uma temporal).

Embora o software tenha sido testado com objetos de geometria simples (dipolos), ele está preparado para tratar problemas complexos (de geometria complexa, com diversos materiais diferentes) sem necessidade de alteração.

Destaca-se que, para tornar realidade o software com as características apresentadas, foi desenvolvida uma nova técnica computacional, a qual foi denominada “organização de variáveis por plano espacial”.

Com a técnica proposta, tornou-se viável a execução de algoritmos que demandem extensiva quantidade de memória, em computadores de uso geral, acessível a qualquer pessoa envolvida em trabalhos de pesquisa científica. Contudo, apesar da otimização conseguida, ficou evidente a redução expressiva da velocidade de execução do algoritmo, quando da necessidade de acesso à disco para suprimir a falta de memória RAM.

Mas, devido ao método de organização dos dados e como o processamento é realizado, os algoritmos e as técnicas de programação propostos permitem a sua fácil adaptação às técnicas de processamento paralelo de dados. Assim, tem-se potencialmente eliminada as limitações associadas à memória RAM.

Além disso, percebeu-se que a técnica de “organização de variáveis por plano espacial” pode ter sua aplicação estendida para qualquer tipo de algoritmo cuja formulação esteja baseada na discretização do espaço tridimensional em elementos finitos e o cálculo sucessivo das propriedades de interesse ao longo do espaço-tempo. Ou seja, a técnica computacional desenvolvida pode ter um campo de aplicação mais amplo que o objeto específico do trabalho.

Logo, por essa possibilidade, o trabalho apresentado tem valor não apenas pelo software obtido, mas também pela nova técnica computacional desenvolvida (“organização de variáveis por plano espacial”).

Ainda para justificar a validade do trabalho apresentado, diversos outros estudos sobre o mesmo tema foram encontrados em publicações internacionais e na internet [9][10][11], relatando o sucesso de várias implementações. Porém, percebeu-se que essas soluções são de caráter específico e direcionadas para a solução de um dado caso ou problema.

Aliados a isso, técnicas complementares já permitem a avaliar as interações de longa distância, a partir dos resultados obtidos através do método FDTD no estudo do espaço no entorno dos objetos de interesse [8]. Embora essas técnicas complementares não sejam temas de nosso trabalho, pois seu estudo isoladamente resultaria em uma outra dissertação de mestrado, elas abrem a perspectiva para a não necessidade de empregar o método FDTD na totalidade do espaço a ser estudado (apenas na região de campo próximo), o que implica em economia de recursos computacionais e tempo.

Finalmente, percebeu-se que a implementação de uma interface gráfica ao software proposto e a realização de um maior número de simulações poderão conduzir a obtenção de uma ferramenta FDTD de uso genérico e com potencial comercial.

## Referências Bibliográficas e Outras

- [1] Allen Taflove e Susan C. Hagens, "Computational Electrodynamics: The Finite Difference Time- Domain Method", 2<sup>nd</sup> Edition, Artech House, Cap 2 – 8, pp. 35-372, Jun 2000.
- [2] Warren L. Stutzman e Gary A. Thiele, "Antenna Theory and Design", 2<sup>nd</sup> Edition, John Willy & Sons, Cap 11, pp. 493-543, Dez 1997.
- [3] Kunio Sawaya, "Numerical Techniques for Analysis of Eletromagnetic Problems", IEICE Trans. Commun., vol. 3, pp. 449-451, Mar 2000
- [4] Kane S. Yee, "Numerical Solutions of Initial Boundary Value Problems Involving Maxwell Equations in Isotropic Media", *IEEE Transactions on Antennas and Propagation*, vol.14, pp 302-307, 1966
- [5] Liao, Z. P., H. L. Wong, B. P. Yang and Y. F. Yuan, "A Transmitting boundary for transient wave analyses", *Scientia Sinica (series A)*, vol. XXVII, pp 1063-1076, 1984
- [6] Berenger, J. P., "A Perfectly matched layer for the FDTD solution of wave-structure interaction problems", *J. Computational Physics*, vol. 114, pp 185-200, 1994
- [7] Sacks, Z. S., D. M. Kingsland, R. Lee, and J. F. Lee, "A perfect matched anisotropic absorber for use as an absorbing boundary condition", *IEEE Transactions on Antennas and Propagation*, vol. 43, pp 1460-1463, March, 1995
- [8] K. S. Yee, D. Ingham, and K. Shlager, "Time-Domain Extrapolation to the Far Field Based on FDTD Calculations", *IEEE Transactions on Antennas and Propagation*, vol. 39, no. 3, pp 410-413, March, 1991
- [9] J. G. Maloney, K. L. Shlager, and G. S. Smith, "A Simple FDTD Model for Transient Excitation of Antennas by Transmission Lines", *IEEE Transactions on Antennas and Propagation*, vol. 42, no. 2, pp 289-292, February, 1994
- [10] D. M. Sullivan, "A Frequency-Dependent FDTD Method for Biological Applications", *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 3, pp 532-539, March, 1992
- [11] Site Oficial do FDTD – <http://www.fDTD.org>
- [12] Fernando C. C. de Castro e P. R. F. Franco, "Antenas", Cap. 1-7, disponíveis na internet, no site <http://diana.ee.pucrs.br/~descatro/download.html>
- [13] Site do compilador DJGPP: [www.delorie.com/djgpp](http://www.delorie.com/djgpp)
- [14] Site do compilador Mingw: [www.mingw.org](http://www.mingw.org)
- [15] Site da IDE Dev-C++: [www.bloodshed.net](http://www.bloodshed.net)
- [16] "Hard Disk Design and Tecnology", <http://www.usbyte.com/common/HDD.htm>
- [17] "Linus Tutorial – The Computer Itself – Hard Disk" <http://www.linux-tutorial.info/modules.php?name=Tutorial&pageid=126>
- [18] Charles Borges de Lima, "Análise de Dispositivos Eletromagnéticos para Hipertermia Usando o Método FDTD", Tese de Doutorado apresentada na UFSC, 2006