

ANÁLISE DE DESEMPENHO DE ALGORITMOS DE OTIMIZAÇÃO PARA EXTRAÇÃO DE PARÂMETROS EM MODELOS SEMICONDUTORES DE POTÊNCIA

Nadil António Sanches Tavares



Departamento de Engenharia Eletrotécnica

Instituto Superior de Engenharia do Porto

2013

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Disciplina de Tese, do 2º ano, do Mestrado em Engenharia Eletrotécnica e de Computadores

Candidato: Nadil António Sanches Tavares, N° 1030844, 1030844@isep.ipp.pt

Orientação científica: Rui Chibante, rmc@isep.ipp.pt

Empresa:

Supervisão: Rui Chibante, rmc@isep.ipp.pt



Departamento de Engenharia Eletrotécnica

Instituto Superior de Engenharia do Porto

27 de Abril de 2013

Aos meus pais, Maria Rosa e Cipriano Tavares, e principalmente a minha filha Maria Eduarda

Agradecimentos

Ao professor Doutor Rui Chibante pela sua orientação e incondicional apoio durante toda esta etapa. Em especial o meu obrigado por, desde sempre, ter acreditado e estimulado a realização deste projeto.

A todos os Colegas do Departamento de Engenharia Eletrotécnica do Instituto Superior de Engenharia pela amizade que se refletiu numa constante disponibilidade, da qual nem sempre soube tirar partido.

Ao Instituto Superior de Engenharia e Departamento da Eletrotécnica, na pessoa dos responsáveis dos seus órgãos de gestão, por me terem proporcionado todas as condições para a realização desta dissertação.

A todos aqueles que direta ou indiretamente se cruzaram no meu percurso, o meu reconhecimento.

A família e aos amigos, dentro e fora do curso, que com uma atitude crítica construtiva, me ajudaram e incentivaram na realização deste trabalho.

Resumo

Apresenta-se nesta tese uma revisão da literatura sobre a modelação de semicondutores de potência baseada na física e posterior análise de desempenho de dois métodos estocásticos, *Particle Swarm Optimizaton (PSO)* e *Simulated Annealing (SA)*, quando utilizado para identificação eficiente de parâmetros de modelos de dispositivos semicondutores de potência, baseado na física. O conhecimento dos valores destes parâmetros, para cada dispositivo, é fundamental para uma simulação precisa do comportamento dinâmico do semicondutor. Os parâmetros são extraídos passo-a-passo durante simulação transiente e desempenham um papel relevante.

Uma outra abordagem interessante nesta tese relaciona-se com o facto de que nos últimos anos, os métodos de modelação para dispositivos de potência têm emergido, com alta precisão e baixo tempo de execução baseado na Equação de Difusão Ambipolar (EDA) para díodos de potência e implementação no MATLAB numa estratégia de otimização formal.

A equação da EDA é resolvida numericamente sob várias condições de injeções e o modelo é desenvolvido e implementado como um subcircuito no simulador IsSpice. Larguras de camada de depleção, área total do dispositivo, nível de dopagem, entre outras, são alguns dos parâmetros extraídos do modelo.

Extração de parâmetros é uma parte importante de desenvolvimento de modelo. O objetivo de extração de parâmetros e otimização é determinar tais valores de parâmetros de modelo de dispositivo que minimiza as diferenças entre um conjunto de características medidas e resultados obtidos pela simulação de modelo de dispositivo. Este processo de minimização é frequentemente chamado de ajuste de características de modelos para dados de medição. O algoritmo implementado, *PSO* é uma técnica de heurística de otimização promissora, eficiente e recentemente proposta por Kennedy e Eberhart, baseado no comportamento social.

As técnicas propostas são encontradas para serem robustas e capazes de alcançar uma solução que é caracterizada para ser precisa e global.

Comparada com algoritmo SA já realizada, o desempenho da técnica proposta tem sido testado utilizando dados experimentais para extrair parâmetros de dispositivos reais das características I-V medidas.

Para validar o modelo, comparação entre resultados de modelo desenvolvido com um outro modelo já desenvolvido são apresentados.

Palavras-Chave

Simulated Annealing, Particle Swarm Optimization, Pesquisa Heurística, Extração de parâmetros, Semicondutores, Otimização.

Abstract

This thesis presents a review of literature on the modeling of physic-based power semiconductor models and subsequent analysis of two stochastic optimization methods, Particle Swarm Optimization and Simulated Annealing, when used for an efficient identification of model parameters. The knowledge of the values of these parameters for each device is essential for an accurate simulation of dynamic behavior of the semiconductor.

Another interesting approach in this thesis relates to the fact that in recent years, the modeling methods for power devices has emerged, with high accuracy and low runtime based on the Ambipolar diffusion equation (ADE) for power devices and MATLAB implementation in a formal implementation strategy.

ADE is solved numerically under various conditions of injection and the model is developed and implemented as a subcircuit in the IsSpice simulator. Widths of the depletion layer, the total area of the device, doping level, among others, are some of the parameter extracted from the model.

Parameter extraction is an important part for model development. The purpose of parameter extraction and optimization is to determine such model parameter values that minimize the differences between a set of measured characteristics and results obtained by the simulation device model. This minimization process is often called tuning characteristics of models for the measurement data. The implemented PSO algorithm is a promising, efficient and recently proposed approach by Kennedy and Eberhart based on social behavior.

The proposed techniques are found to be robust and able to reach a solution that is characterized to be accurate and global.

Compared with SA algorithm, the performance of the proposed technique has been tested using experimental data to extract device parameters I-V characteristics of real measures.

To validate the model, model results are compared with another model already developed.

Keywords

Simulated Annealing, Particle Swarm Optimization, Heuristic Search, parameter extraction, semiconductor power and optimization algorithm.

Índice

AGRADECIMENTOS.....	VI
RESUMO.....	VII
ABSTRACT	IX
ÍNDICE	XI
ÍNDICE DE FIGURAS	XV
ÍNDICE DE TABELAS	XVIII
ACRÓNIMOS	XX
LISTA DE SÍMBOLOS	XXI
1. INTRODUÇÃO	5
1.1. CONTEXTUALIZAÇÃO	6
1.2. OBJETIVOS.....	8
1.3. ORGANIZAÇÃO DA TESE.....	9
2. MODELAÇÃO DE SEMICONDUTORES.....	11
2.1. INTRODUÇÃO.....	12
2.2. CLASSIFICAÇÃO DOS MODELOS	14
2.2.1 Modelos empíricos.....	15
2.2.2 Modelos analíticos	16
2.2.2.1 Modelos de Lumped-charge.....	18
2.2.2.2 Modelos de transformação de cargas.....	23
2.2.2.2.1 Transformada de Laplace	23
2.2.2.2.2 Função de transformada de Fourier.....	24
2.2.2.3 MODELO DE CARGA DISTRIBUÍDA	28
2.2.3 Modelos numéricos	29
2.2.4 Modelos híbridos.....	30
2.2.4.1 MODELOS HÍBRIDOS: Método de diferenças finitas.....	31
2.2.4.1.1 Discretização temporal.....	31
2.2.4.1.2 Discretização espacial	32
2.2.4.1.3 Modelação.....	33
2.2.4.2 MODELOS HÍBRIDOS: Método de elementos finitos.....	35
2.2.4.2.1 Método Rayleigh-Ritz aplicado a EDA.....	37
2.3. ALGORITMOS DE OTIMIZAÇÃO	39
2.3.1 Técnicas de otimização.....	40

2.3.1.1.	Meta-heurísticas	40
2.3.1.1.1.	Algoritmo Genético.....	41
2.3.1.1.2.	Programação Evolucionária.....	41
2.3.1.1.3.	Pesquisa Tabu.....	41
2.3.1.1.4.	Hill Climbing (Subida de Montanha)	42
2.3.1.1.5.	Colônia de Formigas	42
2.3.1.1.6.	Simulated annealing (Recozimento Simulado).....	43
2.3.1.1.7.	Particle Swarm Optimization (PSO).....	48
2.3.2.	Análise qualitativa dos algoritmos	53
2.3.2.1.	Particle Swarm Optimization.....	54
2.3.2.2.	Simulated annealing	55
2.3.2.3.	Algoritmo Genético.....	56
2.3.2.4.	Colônia de formigas.....	56
2.3.2.5.	Hill Climbing	56
2.3.2.6.	Tabu Search (TS)	57
2.4.	CONCLUSÕES.....	57
3.	MODELAÇÃO DE DÍODOS DE POTÊNCIA BASEADA NO MEF	59
3.1.	NOMENCLATURA.....	59
3.2.	INTRODUÇÃO.....	60
3.3.	MODELO PARA CAMADAS DE BASE (N^-).....	61
3.3.1.	Modelo para a equação de difusão ambipolar	61
3.4.	MODELAÇÃO DAS ZONAS P^+ E N^+	63
3.5.	MODELO PARA AS QUEDAS DE TENSÃO	65
3.5.1.	Modelo relacionado ao cálculo das quedas de tensão V_{JR} e V_{JL}	65
3.5.2.	Modelo relacionado com cálculo de V_{RES}	65
3.5.3.	Modelo relacionado com a queda de tensão na zona de depleção	66
3.5.4.	Cálculo das fronteiras na zona de depleção	67
3.6.	IMPLEMENTAÇÃO NO CÓDIGO FONTE ISSPICE	68
3.6.1.	Circuito relacionado as resistências R_J , R_I e R_{IJ}	68
3.6.2.	Circuito relacionado aos condensadores C_I , C_J e C_{IJ}	70
3.6.3.	Circuito relacionado aos condensadores I_l e I_r	72
3.6.4.	Circuito relacionado com as quedas de tensão V_{JL} e V_{JR}	74
3.6.5.	Circuito relacionado com as quedas de tensão V_{RES}	74
3.6.6.	Circuito relacionado com as quedas de tensão V_{SC}	76
3.6.7.	Circuito associado ao cálculo de X_{SC}	77
3.6.8.	Circuito final obtido.....	79
3.7.	Conclusões	80
4.	EXTRAÇÃO DE PARÂMETROS DO MODELO DO DÍODO DE POTÊNCIA	81
4.1.	PROCEDIMENTO DE EXTRAÇÃO DE PARÂMETROS	82
4.2.	MÉTODOS CLÁSSICOS DE ESTIMAÇÃO DE PARÂMETROS	84
4.3.	ESTIMAÇÃO DAS FORMAS DE ONDA	87

4.4.	MÉTODOS DE ESTIMAÇÃO DE PARÂMETROS POR OTIMIZAÇÃO.....	88
4.4.1.	Particle Swarm Optimization para extração de parâmetros de semicondutores	89
4.4.1.1.	Definição dos parâmetros do PSO	93
4.4.1.2.	Procedimento computacional	94
4.4.1.3.	Resultados de simulações	103
4.4.2.	Simulated Annealing para extração de parâmetros de semicondutores	106
4.4.2.1.	Resultado de Simulações	115
4.4.2.2.	Análise comparativa	116
4.5.	CONCLUSÕES	120
5.	DISCUSSÃO / CONCLUSÃO.....	123
	REFERÊNCIAS DOCUMENTAIS	126

Índice de Figuras

Figura 2. 1 - Formas de onda turn-off do díodo	16
Figura 2. 2 - Concentração da carga móvel no topo 25 μ m de IGBT (ATLAS 2-D) (Palmer et al., 2003)	17
Figura 2. 3 - Taxa de recombinação de lacuna em cm ⁻³ /s no fim do cátodo de IGBT no estado on para $V_g = 15$ V e $V_a = 5$ V	18
Figura 2. 4 - Estrutura de díodos de potência unidimensional indicando localização de cinco nós de carga aglomeradas	20
Figura 2. 5 - Modelo de díodo PIN descrito por redes de duas portas	24
Figura 2. 6 - Representação com malhas RC da equação de difusão de portadores	25
Figura 2. 7 - Esquema para solução análogo de equação de difusão portadores ambipolar	27
Figura 2. 8 - Diagrama de bloco do díodo no Simulink (Palmer et al, 2008)	28
Figura 2. 9 - Nós e elementos finitos num domínio discretizado	35
Figura 2. 10 - Rede RC com n-1 elementos finitos unidimensional	39
Figura 2. 11 - Fluxograma de algoritmo Recozimento Simulado	45
Figura 2. 12 - Simulated annealing: a - Escala de arrefecimento linear $T(t)$ e b – Esboço de de energia média resultante $E(t)$ como uma função de tempo.	46
Figura 2. 13 - Topologia de comunicação das partículas: a) Local e b) Global.....	52
Figura 3. 1 - Circuito elétrico equivalente a um elemento finito.....	62
Figura 3. 2 - Circuito elétrico equivalente da EDA pelo MEF.....	63
Figura 3. 3 - Subcircuito responsável pela simulação de uma resistência variável.....	68
Figura 3. 4 - Subcircuito da capacidade variável	70
Figura 3. 5 - Esquema do circuito utilizado para simulação das zonas emissoras.	72
Figura 3. 6 - Circuito responsável pela simulação das quedas de tensão V_{JL} e V_{JR}	74
Figura 3. 7 - Esquemas de circuitos associados à queda de tensão V_{RES}	75
Figura 3. 8 - Esquema do circuito da queda de tensão na zona de depleção.....	77
Figura 3. 9 - Circuito associado ao cálculo das fronteiras na camada de depleção.....	79
Figura 3. 10 - Modelo final do díodo de potência (Araújo, 98)	80
Figura 4. 1 - Diagrama de procedimento de extração de parâmetros (Palmer, 2003)	84
Figura 4. 2 - Formas de ondas de corrente de recuperação reversa.....	87
Figura 4. 3 - Mecanismo de pesquisa no PSO.....	93
Figura 4.4 - Formas de ondas em turn-off depois de otimização do díodo – experimental e simulado	105
Figura 4. 5 - Evolução da função com iteração.....	105
Figura 4. 6 - Evolução dos parâmetros (NB-a; WB –b) com a iteração.....	106

Figura 4. 7 - Procedimento de Simulated Annealing	114
Figura 4. 8 - Formas de ondas em turn-off depois da otimização do díodo – experimental e simulado (SA).....	116
Figura 4. 9 – Evolução dos parâmetros (NB – a; WB - b) com iteração.....	117
Figura 4. 10 - Evolução da função objetivo com iteração.....	117

Índice de Tabelas

Tabela 2. 1 - Valores máximos possíveis no cálculo de passo de tempo.	35
Tabela 4. 1 - Parâmetros de modelo de díodo (Kraus & Mattausch, 1998)	83
Tabela 4. 2 - Parâmetros extraídos (40 iterações / 30 partículas).....	99
Tabela 4. 3 - Parâmetros extraídos (60 Iterações / 20 partículas).....	100
Tabela 4. 4 - Parâmetros extraídos (80 iterações / 15 partículas).....	101
Tabela 4. 5 - Parâmetros extraídos (100 iterações / 12 partículas).....	102
Tabela 4. 6 -Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 4,0	112
Tabela 4. 7 - Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 4,5	113
Tabela 4. 8 - Tabela 4.8 - Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 4.8	114
Tabela 4. 9 - Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 5.0	115
Tabela 4. 10 - Tabela de análise estatística dos algoritmos PSO e SA com amostra 100.tica dos algoritmos PSO e SA com amostra 100.	118

Acrónimos

CPU	–	<i>Central Processing Unit</i>
CV	–	Coeficiente de Variação
CAD	–	<i>Computer Aided Design</i>
DP	–	Desvio Padrão
DC	–	<i>Direct Current</i>
EDA	–	Equação de Difusão Ambipolar
FO	–	Função Objetivo
GTO		<i>Gate Turn-Off</i>
ICAPS	–	<i>Interactive Circuit Analysis Program Selector</i>
IGBT	–	<i>Insulated-Gate Bipolar Transistor</i>
MATLAB	–	<i>MATrix LABORatory</i>
MDF	–	Método das Diferenças Finitas
MEF	–	Método de Elementos Finitos
PSO	–	<i>Particle Swarm Optimization</i>
SA	–	<i>Simulated Annealing</i>
FE	–	<i>Forward Euler</i>
BE	–	<i>Backward Euler</i>

Lista de Símbolos

A	–	Área	cm^2
Ae	–	Área do elemento e	cm^2
Al, Ar	–	Área relativo ao primeiro e último elemento	cm^2
Cii	–	Capacidade do entre nó i	F
Cij	–	Capacidade mútua dos nós i e j	F
C1,C3	–	Capacidade entre os nós e a referência do diporto elementar	F
C2	–	Capacidade entre nós do diporto elementar	F
Dn, Dp,	–	Coefficientes de difusão de elétrons, lacunas e ambipolar	cm^2/s
Ee	–	Elemento finito e	--
Gii	–	Condutância do nó i	Ω^{-1}
Gij	–	Condutância entre nós i e j	Ω^{-1}
hn e hp	–	Taxa de recombinação e elétrons e lacunas	cm^4/s
Id e Irm	–	Corrente direta e inversa máxima no díodo	A
Ip, In , e IT	–	Correntes de lacunas, elétrons e total	A
Ir ,Il	–	Corrente associados as condições fronteiras a direita e a esquerda	A
Jp, Jn, JT	–	Densidades de correntes de lacunas, elétrons e total	A^2/cm
KT	–	Constante de Boltzmann	$J/^{\circ}K$
LEe	–	Largura de Elemento Finito e	cm
Nb, N (x)	–	Concentração ao longo da base	cm^{-3}
NA, ND	–	Concentração de aceitadores e dadores	cm^{-3}
N_{β}^e	–	Função na forma β do elemento e	--
n, n _n	–	Concentrações de elétrons e concentração de elétrons do tipo n	cm^{-3}
ni	–	Concentração intrínseca	cm^{-3}

1. INTRODUÇÃO

No capítulo da introdução será feita a contextualização temática presente nesta tese. Desta forma, será feito o enquadramento, as motivações, os objetivos genéricos, os capítulos e a respetiva organização.

1.1. CONTEXTUALIZAÇÃO

Áreas como economia, medicina, química, arquitetura, indústria automível, eletrônicas e, muitas outras requerem realidade física para ser simulada. Muitas vezes é necessário a simulação porque a realização de uma experiência real é impossível ou bastante complexa. Neste caso, a opção pelo processo de simulação computacional, mais concreto, a simulação de circuitos eletrônicos, em que é descrito matematicamente o modelo completo do circuito. Portanto, podemos afirmar que a simulação de circuito de eletrônica de potência é um assunto cuja importância não pode ser subestimada, especialmente com novas topologias de controle está a ser atingida rapidamente.

Um número de conceitos novos para aperfeiçoar as equações de físicas básicas para exigências de um modelo de dispositivos semicondutores de potência para simulação de circuito tem sido apresentado. Para muitos projetos e análises de circuitos de eletrônica de potências, os softwares de auxílio nas simulações estão a tornar-se cada vez mais importantes.

Um dos objetivos principais no desenvolvimento do circuito orientado a modelos baseados na física é construir modelos disponíveis, que permitem prever a comutação das formas de ondas, a condução e perdas durante a comutação para uma aplicação dos dispositivos de potência. Numa outra vertente, podemos basear na estimação de perdas de potência como uma grande motivação para ambicionarmos o desenvolvimento de circuito de potência. Uma fonte significativa de tais perdas é comumente a comutação dos dispositivos. Por isso, um modelo preciso e baseado na física é exigido, assim como a simulação dever ser válida por um vasto leque de condições.

Por outro lado, a precisão de modelos depende muito da qualidade dos seus parâmetros. Este assunto que não tem tido atenção suficiente no caso de modelos semicondutores de potência para a simulação de circuitos.

Os modelos mais simples, normalmente têm um conjunto pequeno de parâmetros que são adquiridos fundamentalmente por ajustes das características do dispositivo observado. Já os modelos mais avançados demandam um conjunto maior de parâmetros e, uma vez mais baseiam-se na física, os seus parâmetros dependem sobretudo da estrutura, física e tecnologia dos dispositivos.

Essencialmente, os parâmetros devem ser encontrados para um bom ajuste com dispositivos e circuitos. Para obter uma boa estimativa de perdas de comutação, modelos baseados na física são fundamentais.

O maior desafio no desenvolvimento de modelos para semicondutores de potência é descrever o comportamento dinâmico de excesso de portadores nas zonas fracamente dopadas, que pode ser encontrada em todos esses dispositivos. É neste pressuposto que a contribuição da ferramenta CAD (*Computer Aided Design*) será muito importante na previsão de perdas de potência nos circuitos integrados. O objetivo é sempre desenvolver ferramentas de simulação transparente e minimizar os dados de entrada.

Para alcançar o objetivo supracitada é necessário efetuar melhorias nas metodologias CAD e suas aplicações na iniciação da fase de desenvolvimento de circuito de eletrônica de potência. O emprego da metodologia CAD para além de outras características importantes, auxilia na previsão das funcionalidades e credibilidade de um projeto de circuito de potência específico. Tudo isto denota-se que modelos de dispositivos semicondutores de potências de alta qualidade para simulação de circuito são necessários.

Muitos projetistas de circuitos de eletrônica de potência ainda hesitam em utilizar as ferramentas de simulações de circuitos, embora os modelos apropriados de dispositivos de potência estão atualmente em alguns programas de simulação. Para ultrapassar esta resistência, um *software* de simulação acessível é ambicionado e deve ser de fácil utilização de ponto de vista computacional. Os fabricantes de semicondutores de potência deverão fornecer modelos parametrizados dos seus produtos que podem completar ou até mesmo substituir as fichas técnicas, desde que muito mais informações e precisão estariam disponíveis.

Essa tendência para uma melhoria da metodologia para projetos de circuito de potência assim como a carência da melhoria de modelos de dispositivos semicondutores de potência para simulação de circuito não tem sido apenas reconhecido no meio acadêmico mas também pelas indústrias de *software*.

A extração de parâmetros dos modelos semicondutores de potência desempenha um papel importante na interligação entre fabricantes dos semicondutores e projetos de circuitos integrados. A maioria das ferramentas de otimização e extração comercial tem um procedimento de otimização e extração padrão. O método SA (*Simulated Annealing*)

deriva do seu nome de recozimento na metalurgia que é uma técnica envolvendo o aquecimento e arrefecimento controlado do material para aumentar a energia dos seus cristais para permiti-los mover para cima. Baseado nesta analogia, cada passo de SA do processo físico escolhe a solução atual como uma próxima solução aleatória. Como alternativa, PSO (*Particle Swarm Optimization*) é um algoritmo de pesquisa aleatória com inteligência de enxame, tendo uma grande capacidade de pesquisa nos problemas não lineares. Comparado com *Simulated Annealing*, PSO necessita de poucos parâmetros a ser configurados e teoricamente a sua implementação e compreensão é relativamente mais fácil. Ademais, o PSO tem velocidade de convergência alta. Consequentemente, pode encontrar soluções ótimas rapidamente durante a extração de parâmetros

O procedimento de extração de parâmetros tem de ser razoavelmente simples para ser funcional. Neste trabalho propusemos a implementação e análise de dois algoritmos de otimização promissores na extração de parâmetros semicondutores de potência, que consiste nas medições dos erros entre dados experimentais e simulados, por um procedimento automatizado para aperfeiçoar e aumentar a precisão dos parâmetros extraídos.

Tentativas anteriores nos procedimentos de parametrização automatizada têm concentrado ou em características de montagens DC ou indicadores chaves de performance, como por exemplo, correntes e tempos de recuperação reversa dos díodos de potência (Bryant et al., 2006). Neste âmbito, o estudo realizado nesta dissertação ambiciona suportar o avanço do estado da arte da modelação de dispositivos semicondutores de potência, particularmente, modelação de díodos de potência.

1.2. OBJETIVOS

Os objetivos primários desta tese podem ser resumidos como seguintes:

- Fazer uma revisão da literatura sobre a modelação de semicondutores de potência baseada na física de semicondutores, caracterizando os conjuntos de parâmetros de natureza física e tecnológica.
- Analisar o desempenho do algoritmo PSO e SA para tarefas de simulação e extração de parâmetros de modelos semicondutores de potência baseado na física.

1.3. ORGANIZAÇÃO DA TESE

Esta tese está dividida em cinco partes. No primeiro capítulo apresenta-se a introdução geral do trabalho desenvolvido. No capítulo seguinte apresenta-se o estado de arte relacionado com a modelação dos dispositivos semicondutores de potência e algoritmo de otimização na extração de parâmetros desses semicondutores.

No terceiro capítulo é constituído pelo desenvolvimento do modelo de díodo de potência, caracterizado pelo método de Elemento Finito. Neste capítulo faz-se um mapeamento das várias regiões que constituem a díodo de potência, com principal destaque para a zona da camada de depleção, que é a mais levemente dopada do dispositivo, sendo considerado a de interesse capital neste trabalho. No final deste capítulo é demonstrada a implementação no IsPice4 do díodo de potência, um simulador de uso genérico para circuitos elétricos.

No quarto capítulo é dedicado a nova técnica de estimação dos parâmetros físicos. Neste capítulo faz-se ainda a avaliação de formas de ondas experimentais e simuladas detalhadamente através dos erros quadráticos associados aos parâmetros do dispositivo para cada algoritmo durante a comutação das formas de ondas.

Para finalizar, no quinto e último capítulo, os resultados obtidos são comentados e sintetizadas em conclusões gerais e as perspectivas para futuro trabalho.

2. MODELAÇÃO DE SEMICONDUTORES

Nesta secção efetua-se o estado da arte ligada à modelação de dispositivos semicondutores de potência e análise de alguns algoritmos de otimização para extração de parâmetros de referidos modelos, com ênfase para os díodos de potência e transístores IGBTs.

Nos últimos anos, pesquisas de novos modelos semicondutores de potência têm aumentado. Vários são os programas de simulação de circuitos eletrónicos propostos, como por exemplo o SABER e o SPICE.

Neste capítulo é feita uma caracterização dos vários tipos de modelos para ajudar a perceber o que é a modelação e simulação dos semicondutores.

2.1. INTRODUÇÃO

Modelar componentes, especialmente semicondutores, é uma atividade fascinante e importante (Leventhal & Green, 2006a). Isto pode iniciar nos níveis de energia atômico e estender-se os efeitos da saída de comutação de energia em emissões radiadas de um invólucro.

A modelação dos semicondutores iniciou nos anos cinquenta, quando o Van Roosbroeck formulou as equações tendência-difusão (Roosbroeck, 1950) e mais tarde por físicos como (Shockley & Read, 1952).

Nas últimas quatro décadas, a modelação matemática e simulação de transporte de carga nos semicondutores tornou-se uma área de pesquisa próspera em matemática aplicada.

Em (Kraus & Mattausch, 1998a) foram propostos vários novos conceitos para sustentar as equações físicas básicas para as exigências de um modelo de dispositivo semicondutor para a simulação.

O estímulo especial no desenvolvimento de modelos para simulação de circuitos origina na necessidade concomitante nas contradições requeridas na elevada precisão e demanda de esforço do calculo computacional e, modelos de parâmetros de fácil acesso a fim de colocar ponto final no conflito de interesse favorável desses requisitos.

No extremo oposto, as condições de contornos económicos geral limitado, demandam uma melhor eficiência e eficácia na simulação e análise de circuitos. Deste modo um progresso apenas pode ser alcançado por meio de atualização de programas de projeto assistido por computador (em inglês CAD) e sua aplicação no campo de eletrónica de potência. Estes programas também auxiliam engenheiros de projetos dispositivos semicondutores e engenheiros de processos ao projetar estruturas, materiais e processamentos que produzem as propriedades elétricas desejadas.

As ferramentas de análise de circuitos de potência têm empregado modelos de semicondutores muito simples. No processo de modelação é exigido uma análise de simulação de dispositivo semicondutor em que a finalidade é citar o método no qual o problema complexo que caracteriza o dispositivo é transformado num problema mais simples.

Naturalmente que com poder computacional cada vez maior e o desenvolvimento de ferramentas de simulações rápidas tem contribuído para um crescente interesse nos projetos ótimos de dispositivos semicondutores.

Muitos projetistas nesta área, ainda hesitam na utilização de ferramentas de simulação, embora os modelos adequados de dispositivos de potência já estejam disponíveis para alguns programas de simulação. Portanto, para ultrapassar este obstáculo, é desejado um software de baixo custo e fácil manuseamento. Modelos parametrizados dos produtos devem ser fornecidos pelos fabricantes de semicondutores a fim de complementar ou mesmo substituir as fichas técnicas uma vez que estaria disponível informação em maior quantidade e com mais precisão.

A modelação muitas vezes enfrenta uma série de desafios, abrangendo a complexidade, dinâmica de um sistema ou processo, cálculo numérico, não-linearidade, incertezas tais como erros sistemáticos e estocásticos, insuficiência de dados e tempos, variáveis de sistemas ocultas, fraca compreensão da verdadeira natureza do sistema em questão.

Entre os dispositivos de potência, os modelos mais avançados, e com maior precisão são incorporados nos programas de simulação, que permitem uma inserção fácil das equações do modelo (Leventhal & Green, 2006b). Esses programas (Eldo e Saber) são muito poderosos porém, também são muito caros. O programa de simulação padrão com uma distribuição larga, no entanto, é o SPICE com os seus numerosos derivados. Sendo um software relativamente barato e de grande utilização, existe uma grande requisição para modelos de dispositivos adequados. Em contrapartida, o método para incorporação de novos modelos no SPICE é restrito e não tão fácil como no Saber. Por causa de complexidade de modelos de circuitos realizada no SPICE, para dispositivos de potência, existe muito consumo do tempo de computação.

Neste capítulo, é realizado um estudo de abordagens diversas relacionadas com modelação de semicondutores com especial destaque para díodos e IGBTs. A modelação é especialmente importante para o fabrico de semicondutores modernos. Devido à elevada complexidade e risco económico, componentes bem modelados desempenham um papel fundamental no diagnóstico de equipamentos e controle do processo (Mattausch, 1998). Neste capítulo dá-se ênfase a modelação de semicondutores em particular, à modelação física e empírica e, por fim técnicas de estimação de parâmetros.

2.2. CLASSIFICAÇÃO DOS MODELOS

Os modelos são essenciais para o *software* de simulação de circuitos cuja finalidade é fazer previsões sobre o comportamento do circuito. Especialmente, esses programas de simulação precisam de modelos de componentes, módulos e subsistemas.

Classificar os modelos de semicondutores não é uma tarefa fácil devido à grande variedade de modelos e diferentes terminologias encontradas na literatura.

Partindo do princípio que modelos são representações de tais realidades físicas, precisaremos conhecer como eles são precisos, e quão precisos eles precisam ser. Um modelo pode ser menos preciso e detalhado do que o item real e usufruir de uma gama mais limitada de condições sobre os quais não é válido. Um bom modelo não necessita de ter uma descrição completa da realidade, mas deve capturar certas propriedades e características relativas a um objetivo específico, como diagnóstico, previsão e classificação. A previsão é frequentemente desejada num bom modelo, porém, dependendo da natureza exata do problema, compromissos entre precisão, simplicidade, eficiência e robustez tem que ser considerados.

Existe muitos desacordos na escolha do melhor modelo para semicondutores, visto que, cada autor defende o seu modelo com entusiasmo e teorias profundas. Estes autores argumentam que o modelo fornece suporte a todos os projetos eletrônicos, para todos os utilizadores e funciona melhor que todos os outros. Por outro lado alguns modelos defendidos perdem credencial pelo facto de apresentarem pouca eficiência em comparação com um modelo alternativo.

Dependendo da natureza discriminada de um modelo, podem ser usadas técnicas distintas de modelação, em que se destacam um modelo físico *versus* um modelo empírico, um modelo paramétrico *versus* um modelo não paramétrico. Contudo, há algumas vertentes comuns em todas as modelações, que são observação e experimentação, modelo de seleção e estimação de parâmetros e modelos de verificação e avaliação.

Portanto, neste capítulo, centra-se a análise nas características importantes da modelação para simulação de semicondutores. Embora existam algumas práticas em comum na modelação de semicondutores, dá-se uma atenção especial à modelação empírica, e modelação física e técnicas para estimação de parâmetros.

2.2.1 Modelos empíricos

Este modelo trata o dispositivo como uma “caixa negra” e o seu comportamento é descrito externamente sem detalhes considerados para efeitos físicos no interior. Portanto, os seus parâmetros normalmente são obtidos por intermédio de tabelas de medidas ou por via de ajuste de curvas (*curve-fitting* em inglês). Este método é o mais empregado, uma vez que consiste na construção da função matemática mais adequada para a séries de dados, que por sua vez relaciona com o dito comportamento externo do dispositivo. Este comportamento externo é geralmente modelado por meio de equações empíricas de ajuste de dados, tabelas de pesquisa ou um subcircuito elétrico de componentes comuns para emular dados experimentais conhecidos (Tan & Tseng, 1999).

A modelação não é obtida por derivações rigorosas da física do dispositivo e é mais apropriado para modelar dispositivos com característica DC. Também muito difícil utilizar para efeitos dinâmicos de dispositivo em ambiente de diferentes circuitos assim como dificilmente conseguirá simular alguns fenómenos de alta tensão.

A aproximação deste modelo trata os dispositivos como uma “caixa negra” e descreve o comportamento observado externamente sem uma consideração detalhada de efeitos físicos ocorrendo no interior do modelo. Se isso acontecer, tensões e correntes nos terminais desses dispositivos são aproximadas diretamente por funções simples. Do lado contrario, é necessário incluir variáveis adicionais dentro da definição da equação. Essas variáveis podem ser (não restrito) variáveis internas dos dispositivos como por exemplo a carga.

As relações entre correntes e tensões são descritas por funções matemáticas que são alcançadas fundamentalmente por suposições intuitivas. A relação é utilizada para aproximar a recuperação inversa de díodos de potência. O parâmetro τ_R é um constante de tempo com responsabilidade de determinar o quão rápido a corrente inversa *turn-off* do díodo diminui (ver Figura 2.1).

$$Q = -I(t)\tau_R \quad (2.1)$$

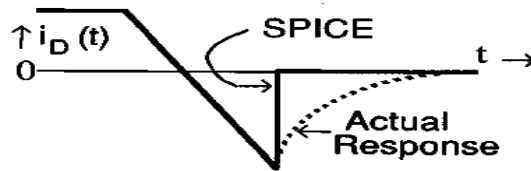


Figura 2. 1 - Formas de onda *turn-off* do diodo

A solução para a corrente é obtida com esta relação e a equação de controlo de carga simplificado. Todavia, esta solução é apenas valida para a fase de turn-off, quando a corrente diminui o seu pico reverso para zero. Outras fases de comutação deve ser descrito por outras funções, e as soluções de diferentes fases dever ser ajustado para garantir a continuidade.

2.2.2. Modelos analíticos

O comportamento elétrico de alguns dispositivos pode ser frequentemente simulado por simuladores baseados na física dos semicondutores mas, para muitas estimativas, modelos analíticos compactos e eficientes são uma solução com melhor relação custo – eficiência.

Neste contexto, uma das mais importantes equações para descrever o comportamento dinâmico do dispositivo é a Equação de Difusão Ambipolar (EDA) que fornece a distribuição dos portadores de carga na região de armazenamento n- do dispositivo.

Um modelo semi-empírico baseado na física dos semicondutores que utiliza uma solução analítica da (EDA) foi proposto por Kraus e Hoffmann (1993).

Em (Carvalho & Araújo, 2004,) apresenta-se uma nova abordagem para a solução da EDA por via de uma formulação variacional e elementos finitos simplex. Esta abordagem foi aplicada em modelos para transístores bipolares de junção (TBJ) e díodos p-i-n.

Em zonas fracamente dopadas do dispositivo, para condições de injeção de alto nível de dopagem, a distribuição de portadores de carga é descrita pela EDA,

$$\frac{\partial p}{\partial t} = D \frac{\partial^2 p}{\partial x^2} - \frac{p}{\tau} \quad (2.2)$$

em que, p é a concentração de lacunas e eletrões, τ é o tempo médio de vida dos portadores e D é a constante de difusão ambipolar.

$$D = 2 \frac{D_p D_n}{D_p + D_n} \quad (2.3)$$

A condição limite associada com a equação 2.2 é:

$$\frac{dp}{dx} = \frac{1}{2qA} \left(\frac{InDp - IpDn}{DnDp} \right) \quad (2.4)$$

Onde I_p / I_n são correntes de lacunas e elétrons, respetivamente, para as fronteiras da região n-.

Em (2.3) e (2.4), D_p e D_n também são constantes de difusão, A é a área do dispositivo e q é a carga do elétron. Partindo da equação (2.2) há um consenso que esta equação surge das equações de continuidade de corrente e densidade de corrente nas condições de quasi-neutralidade e injeção de alto nível ($p \approx n \approx n_i$), nos semicondutores intrínsecos. A solução da EDA obtém-se geralmente por considerar que a distribuição de portadores de carga é unidimensional (1-D) em toda região n- (Palmer et al, 2003). Com a finalidade de assimilar efeitos importantes em modelo bidimensional (2-D) e verificar a precisão de aproximação 1-D foi utilizado simulações 2-D que deve ser conservada no intuito de acautelar uma boa precisão. Como se pode observar na Figura 2.1, o comportamento do dispositivo de potência modelado, particularmente o díodo e IGBTs, são fortemente dependentes de excesso de distribuição de portadores (carga) na região deriva elevada.

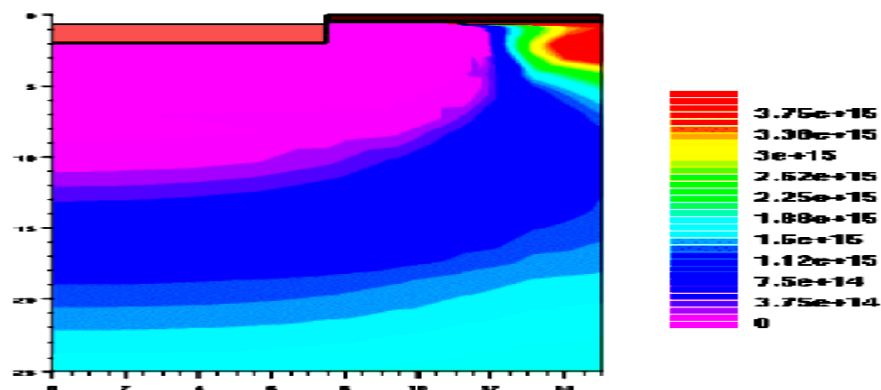


Figura 2. 2 - Concentração da carga móvel no topo 25 μ m de IGBT (ATLAS 2-D) (Palmer et al., 2003)

O perfil da carga apresenta formalmente um 1-D numa totalidade de 90% do volume do IGBT moderno. Podemos ver que na variação 2-D é aproximadamente limitado ao 20 μ m na parte superior do dispositivo. A distribuição da carga na região deriva restante é basicamente unidimensional excetuando a direção vertical que apresenta variações. Contudo, os efeitos 2-D evidenciam mais na zona próxima da porta, onde se realiza o

processamento de transporte de cargas no canal MOS. Como podemos ver na Figura 2.3 a taxa de recombinação diminui na direção de cátodo.

No entanto, geralmente, a EDA não ostenta uma solução analítica exata, pelo que são utilizadas diversas abordagens para chegar à respetiva solução. A partir desta aproximação surgiram diversos modelos divididos em três categorias:

Modelos de Lumped-charge (2.2.2.1)

Modelos de transformação de cargas (2.2.2.2)

Modelos de carga distribuída (2.2.2.3)

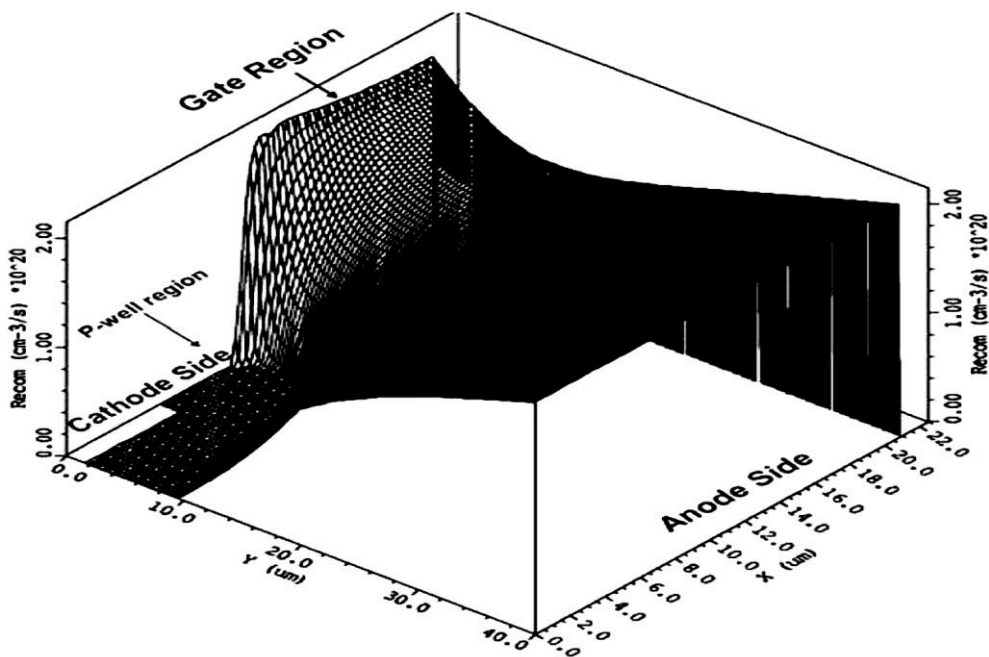


Figura 2.3 - Taxa de recombinação de lacuna em cm⁻³/s no fim do cátodo de IGBT no estado on para $V_g = 15 \text{ V}$ e $V_a = 5 \text{ V}$

2.2.2.1. Modelos de Lumped-charge

A técnica de modelação Lumped-charge representa uma técnica sistemática para modelação analítica. O diódo, retificador controlado de silício (SCR), Gate turn-off (GTO)

Tiristor controlado por MOS (MCT) e modelos MOSFET têm sido criados. Portanto, esta técnica permite uma redução significativa da complexidade do modelo enquanto retém a informação básica sobre o processo de transporte de portadores no dispositivo. O modelo resultante contém um conjunto de equações contínuas que descreve o comportamento dinâmico e estático.

Características externas dos dispositivos semicondutores são diretamente ligadas para a distribuição de portadores interna e suas características de transportes. A técnica de modelação Lumped-charge utiliza as seguintes diretrizes para simplificar as equações semicondutores fundamentais, que determina o comportamento dos elétrons e lacunas (Hartmann & Rieger, 2002).

A estrutura do dispositivo é discretizada para diversas regiões tipo N ou P, cada um dos quais contém armazenamento de carga e nós de conexão. Um nó de armazenamento de carga é responsável pelo armazenamento de portadores de carga e recombinação e é normalmente localizado próximo do centro de uma região. Um nó de conexão liga tensão de junção para o nível de concentração portador-carga e está localizado na borda de depleção-junção.

Os valores de carga de elétron e lacuna em cada nó são obtidos pela multiplicação da concentração portador-carga local pelo volume de região de referência, que é tipicamente uma das regiões fracamente dopadas.

Por fim, as cargas possuem ligações de nós através de seis equações de física de dispositivos e teoria de circuitos:

- equação de densidade de corrente
- equação de continuidade de corrente
- equação de neutralidade de carga
- relações de Boltzman (equações da junção p-n)
- equação de Poisson
- leis de tensão e corrente Kirchoff

Nas Figura 2.4 estão identificadas cinco regiões verticais críticas no díodo, onde podemos também assinalar as cinco equações nesta mesma região. Cada nó de carga acumula a sua carga total em cada região. As regiões fracamente dopadas N- onde se determina a maioria das características dos dispositivos requerem três nós de cargas. Por outro lado, nas zonas fortemente dopadas como no caso de N+ e P+ a distribuição das cargas são importantes apenas nas proximidades de junções, assim é necessário apenas um nó de carga em cada. Cargas de eletrão e lacuna são consideradas separadamente.

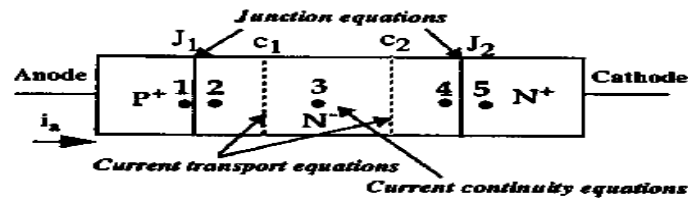


Figura 2. 4 - Estrutura de díodos de potência unidimensional indicando localização de cinco nós de carga aglomeradas

Neste modelo, utilizamos o díodo p-i-n para descrever a técnica Lumped-charge. No entanto, as equações para este dispositivo unidimensional podem ser resumidas nas seguintes equações fundamentais (Lauritzen, 1997):

Equações de transporte de corrente:

$$i_{p23} = \frac{q_{p2} - q_{p3}}{bT} + \frac{q_{p3}}{2bT} \frac{qV_{23}}{2b\phi_t} \quad (2.5)$$

$$i_{n23} = \frac{q_{n3} - q_{n2}}{T} + \frac{q_{n3}}{2T} \frac{V_{23}}{\phi_t} \quad (2.6)$$

$$i_{p34} = \frac{q_{p3} - q_{p4}}{bT} + \frac{q_{p3}}{2bT} \frac{V_{34}}{\phi_t} \quad (2.7)$$

$$i_{n34} = \frac{q_{n4} - q_{n3}}{T} + \frac{q_{n3}}{2T} \frac{V_{34}}{\phi_t} \quad (2.8)$$

Equações de continuidade de corrente:

$$i_{n23} = \frac{q_{n1}}{\tau_1} \quad (2.9)$$

$$i_{p23} - i_{p34} = \frac{q_{p3} - Q_{p0}}{\tau_3} + \frac{dq_{p3}}{dt} \quad (2.10)$$

$$i_{p34} = \frac{q_{p5}}{\tau_5} \quad (2.11)$$

$$i_{n23} = \frac{q_{n1}}{\tau_1} \quad (2.12)$$

$$i_{p23} - i_{p34} = \frac{q_{p3} - Q_{p0}}{\tau_3} + \frac{dq_{p3}}{dt} \quad (2.13)$$

$$i_{p34} = \frac{q_{p5}}{\tau_5} \quad (2.14)$$

Equações de junção P-N:

$$q_{p2} = Q_{p0} e^{\left(\frac{V_{12}}{\phi_t}\right)} \quad (2.15)$$

$$q_{n1} = q_{n2} e^{\left(\frac{V_{12} - \phi_{12}}{\phi_t}\right)} \quad (2.16)$$

$$q_{n4} = Q_B e^{\left(\frac{V_{45}}{\phi_t}\right)} \quad (2.17)$$

$$q_{p5} = q_{p4} e^{\left(\frac{V_{45} - \phi_{45}}{\phi_t}\right)} \quad (2.18)$$

Restrições de terminais (KVL, KCL):

$$V_{AK} = v_{12} + v_{23} + v_{34} + v_{45} \quad (2.19)$$

$$i_A = i_{n23} + i_{p23} \quad (2.20)$$

$$i_A = i_{n34} + i_{p34} \quad (2.21)$$

Equação de neutralidade de carga:

$$q_{n2} = Q_B + q_{p2} \quad (2.22)$$

$$q_{n3} = Q_B + q_{p3} \quad (2.23)$$

$$q_{n4} = Q_B + q_{p4} \quad (2.24)$$

Onde, i_{p23} é a corrente total da lacuna, $q_{p2} - q_{p3}$ constituem cargas nas lacunas nos nós 2 e 3, v_{23} representa a queda de tensão nos nós 2 e 3 e por fim, ϕ_t a tensão térmica.

O parâmetro Q_B equação é a carga de elétron de equilíbrio, v_{45} é a queda de tensão entre os pontos 4 e 5 na região N-.

Nas equações de continuidade de corrente, i_{p23} e i_{p34} são componentes de corrente de lacunas nas extremidades do nó 3 enquanto Q_{p0} é carga de lacuna em equilíbrio nesta equação como nas equações de junção P-N. Já o parâmetro τ é o tempo de vida de lacunas e, q_p e q_n representam a cargas de lacunas e elétrons respectivamente.

Corrente de elétrons e lacunas através da secção c1 é descrito pelas equações (2.5) (2.6) que manifestam da mesma forma. As equações (2.7) e (2.8) descrevem as correntes de lacunas e elétrons na secção c2. Aqui o parâmetro T é o tempo de trânsito de elétron e b, a taxa de elétron por mobilidade de lacuna.

Na equação (2.13), τ_3 é o mesmo para elétron e lacuna, a equação de continuidade de elétron é redundante e assim omitida. A equação (2.12) e (2.14) indicada a corrente de recombinação nos finais de regiões fortemente dopado, onde τ_1 e τ_5 são muito menores que τ_3 .

A equação (2.17) é a equação da junção J2 e a equação (2.15) é a equação correspondente para J1. Equações (2.16) e (2.18) são equações da junção p-n que relacionam as cargas em ambos os lados da junção J1 e J2 onde ϕ_{45} e ϕ_{12} são construídos nas potenciais das duas junções.

As equações (2.19) e (2.21) são expressões de interface que relacionam variáveis internas para variáveis externas V_{AK} , a tensão ânodo – cátodo e i_A , a corrente do ânodo.

As equações (2.22) e (2.24), na equação de neutralidade de carga, expressam as cargas de elétrons e lacunas na região v-.

Os modelos Lumped-charge apresentam um compromisso interessantes entre tempo de simulação precisão. O conceito deste modelo tem demonstrado ser capaz de modelação, pelo menos com um degrau moderado de precisão, as características importantes de fenómeno físico básico, ocorrendo especialmente nos dispositivos semicondutores de potência. Esta abordagem requer poucos parâmetros, que estão em relação estreita com características elétricas do dispositivo. Além disso, a determinação de parâmetros é considerada simples e direta. Os autores (Kraus e Mattausch, 1998b), afirmam que

modelos lumped são novas direções importantes de modelação de dispositivos semicondutores, que se mantêm na expectativa para desvendar desenvolvimentos futuros consideráveis e que sua força principal reside na simulação de circuitos eletrónicos de potência com um número elevado de componentes.

2.2.2.2. Modelos de transformação de cargas

Na modelação analítica de dispositivos semicondutores, são variadas as técnicas matemáticas para resolução de equações diferenciais, nos quais, a título de exemplo, a equação diferencial pode ser transformada numa equação integral. Este modelo é conhecido no meio científico por utilizar a transformada de Laplace e aplicação de transformada de Fourier. Inicialmente estes métodos podem conduzir as soluções exatas, contudo, existem limites relacionados com as condições de contorno e o resultado consiste em séries infinitas. Portanto, neste modelo as soluções são aproximadas, já que para obter os resultados a série tem que ser truncada.

2.2.2.2.1. Transformada de Laplace

Ainda, baseado nos artigos dos autores Kraus e Mattausch (1998c), foi demonstrado que nas técnicas de transformada de Laplace, é substituído a equação de difusão dentro do “espaço s ”.

$$D \frac{d^2 p'}{dx^2} = p' \left(s + \frac{1}{\tau} \right). \quad (2.25)$$

Onde $p' = p(x, t) - p(x, 0)$. Nesta equação, a solução que é convertendo-a de volta para o domínio de tempo consiste numa série de Fourier. Na equação geral (2.25), substituímos $p'(0, s)$ e $p'(W, s)$ nisto, podemos interpretar a equação derivada como equação de uma rede de duas portas, exibido na Figura 2.5 em que os nós de tensão correspondem às concentrações de portadores enquanto as correntes de entrada e saída corresponde às concentrações de portadores derivativos de x . Ainda na equação 2.5, as condições limites não-linear da EDA são representadas por dois geradores de correntes não linear, GE e GD , controlado por corrente através de díodos, $p'(0)$ e $p'(W)$. No circuito em baixo, Z_m e Y_p são função de “ s ” e é aproximada com funções racionais. Função Z_m está relacionada com os dois primeiros termos da série de Taylor e por esta via é representado por séries de resistência R_m e por uma indutância L_m . Por outro lado a função Y_p está relacionada pela

utilização de uma função racional onde coeficientes da mesma são obtidos por utilização de aproximação de Padé¹. A representação aproximada de admitância é alcançada pela manipulação algébrica. Na rede de duas portas podemos constatar que todos os componentes passivos são dependentes de largura epitaxial W , que mantém inalterado durante as simulações. Podemos obter resultados mais precisos por inserção de mais elementos como apontado na Figura 2.5 conectado em série, cuja finalidade é obter concentração de portadores nas outras partes da camada epitaxial. Porém, isto afeta a simulação.

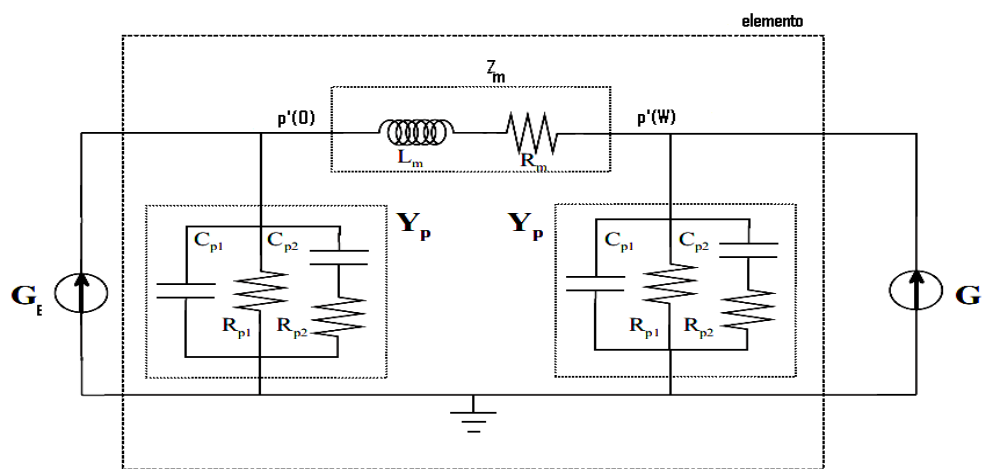


Figura 2. 5 - Modelo de díodo PIN descrito por redes de duas portas

2.2.2.2. Função de transformada de Fourier

O modelo analítico baseado na transformada de Fourier foi proposto por Leturq et al. (1995a) e posteriormente desenvolvido por Bryant et al. (2005) com a finalidade de incluir a variável tempo de vida na zona de depleção, no qual a resolução da equação de EDA é realizada através da solução de Fourier. Varias abordagens feita pelos autores, partiram desta solução para resolver a equação de EDA (Palmer et al., 2003), (R.Kraus, Turkes, & J.Sigg, 1998).

Esta solução também pode ser implementada por analogia elétrica. Leturq et al. (1995b) considerou que partindo da equação básica da nossa bem conhecida equação de difusão

¹ Aproximação de Padé é a melhor aproximação de uma função por uma função racional de uma dada ordem.

(EDA), onde o alto nível de tempo de vida dos portadores é encarado como uma constante de toda camada epilayer, é demonstrada aqui como transformada discreta de Fourier de $p(x,t)$:

$$p(x,t) = v_0(t) + \sum_{k=1}^{\infty} v_k(t) \cos \left[\frac{k\pi(x-x_1)}{x_2-x_1} \right] \quad (2.26)$$

Permite converter a equação EDA (2.22) dentro de um sistema infinito de equações diferenciais linear de primeira ordem em coeficientes de $v_0 \dots, v_k$:

$$(x - x_1) \left(\frac{dv_0}{dt} + \frac{v_0}{\tau} \right) \left[\frac{\partial p}{\partial x} \Big|_{x_2} - \frac{\partial p}{\partial x} \Big|_{x_1} \right] - I_0 \quad (2.27)$$

com:

$$I_0 = \sum_{n=1}^{\infty} v_n \left(\frac{dx_1}{dt} - (-1)^n \frac{dx_2}{dt} \right) \quad (2.28)$$

$$\frac{x_2-x_1}{2} \left(\frac{dv_k}{dt} + v_k \left[\frac{1}{\tau} + \frac{Dk^2\pi^2}{(x_2-x_1)^2} \right] \right) = D \left[(-1)^k \frac{\partial p}{\partial x} \Big|_{x_2} - \frac{\partial p}{\partial x} \Big|_{x_1} \right] - I_k \quad (2.29)$$

Onde,

$$I_k = \frac{v_k}{4} \frac{d(x_1-x_2)}{dt} + \sum_{n=1}^{\infty} \frac{n^2 v_n}{n^2-k^2} \left(\frac{dx_1}{dt} - (-1)^{k+n} \frac{dx_2}{dt} \right) \quad (2.30)$$

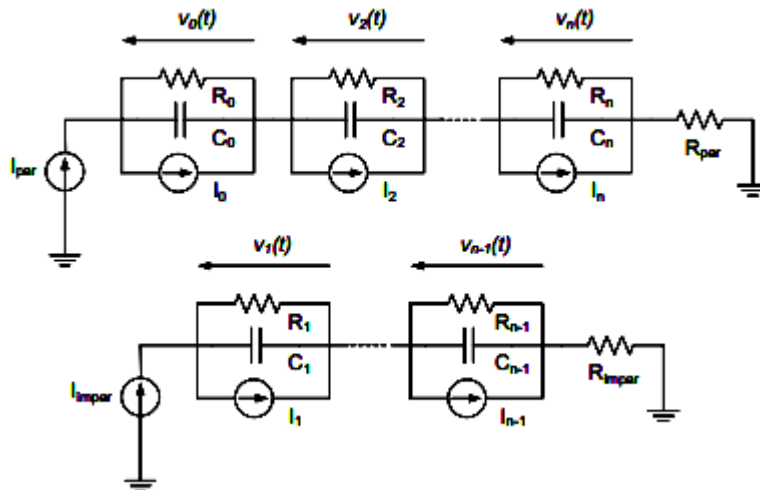


Figura 2. 6 - Representação com malhas RC da equação de difusão de portadores

A Figura 2.6 faz a representação das definições das equações acima, como uma analogia elétrica simples, na forma de duas linhas RC que corresponde a valores pares e ímpares de k . As tensões v_k através de células contínuas que representam os coeficientes de séries de Fourier.

Como se pode constatar na Figura 2.6, as malhas são formadas aos pares por componentes RC, que por sua vez são função da largura da zona de armazenamento de carga $x_2 - x_1$:

$$\text{se } k = 0 \rightarrow \begin{cases} C_0 = x_2 - x_1 \\ R_0 = \frac{\tau}{x_2 - x_1} \end{cases} \quad (2.31)$$

$$\text{se } k \neq 0 \rightarrow \begin{cases} C_k = \frac{x_2 - x_1}{2} \\ R_k = \frac{2}{x_2 - x_1} \frac{1}{\frac{1}{\tau} + \frac{k^2 \pi^2 D}{x_2 - x_1}} \end{cases} \quad (2.32)$$

I_k e I_0 são estão representado nas equações (2.28) e (2.30). As linhas RC são alimentadas por fontes de correntes I_{par} e $I_{\text{ímpar}}$ em que os valores são definidos pelas condições fronteira:

$$I_{\text{par}} = D \left(\left. \frac{\partial p}{\partial x} \right|_{x_2} - \left. \frac{\partial p}{\partial x} \right|_{x_1} \right) \quad (2.33)$$

$$I_{\text{ímpar}} = -D \left(\left. \frac{\partial p}{\partial x} \right|_{x_2} + \left. \frac{\partial p}{\partial x} \right|_{x_1} \right) \quad (2.34)$$

A equação (2.26) é uma série infinita, e pragmaticamente a série é truncada de forma que a constante de tempo da última malha seja inferior à escala de tempo necessária para observar o fenómeno. O erro de truncatura pode ser amplamente reduzido por terminação das linhas, em que adicionam-se as resistências séries respectivas aos valores das resistências armazenadas nas células em falta (ver Figura 2.6). Esta implementação possibilita, o modelo de Fourier conceber internamente o comportamento do dispositivo examinando a distribuição da concentração dos portadores na região da base.

As malhas RC são integradas num bloco computacional ilustrado na figura 27a. As saídas são densidades de portadores, especialmente os valores p_{x1} p_{x2} , nos limites e algumas quantidades de interesses tais como portadoras de cargas armazenadas Q_s . Na figura acima

(2.7a) W , representa a largura de região de base. Nas extremidades das regiões de armazenamento de portadores, onde as regiões são fixadas, os cálculos são efetuados de forma evidente. Na recuperação, as condições limites tornariam móvel, como aparecem nas zonas vazias. Neste caso, os valores das abscissas limites x_1 e x_2 devem ser controlado de forma a manter p_{x1} e/ou p_{x2} no cálculo. Isto é obtido por amplificadores operacionais fictícios realizados por circuitos simples do tipo exibido na figura 27b, o diodo ideal D permitindo a continuidade da representação ser suportado entre os casos de contorno fixo e móvel.

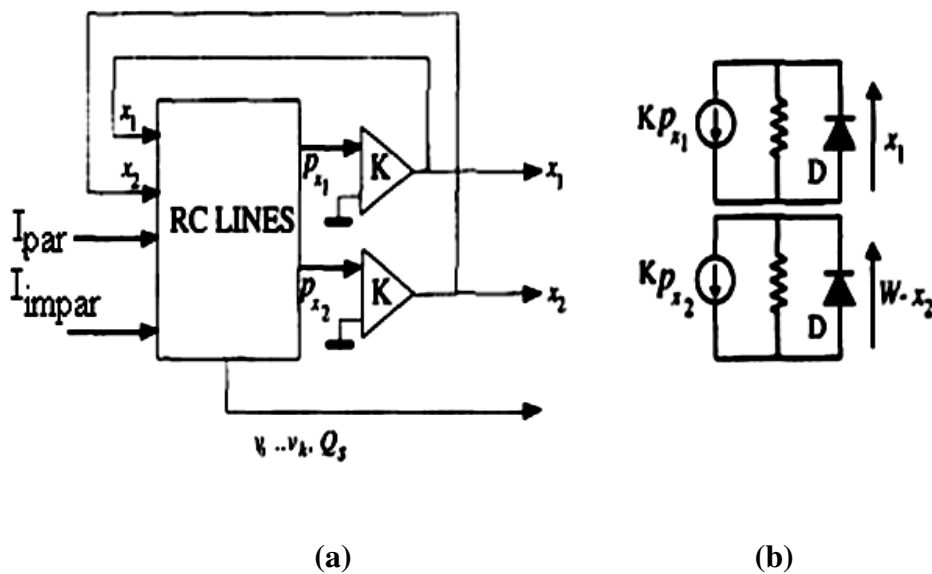


Figura 2. 7 - Esquema para solução análogo de equação de difusão portadores ambipolar

A quantidade de células para ser retida nas linhas RC devem ser tais que a constante de tempo k-ésimo é muito menor que tempos característicos envolvido nas formas de ondas de corrente e tensão.

A desvantagem dos modelos baseados nesta aproximação é que todos os elementos da rede RC são dependentes da largura da região submersa com excesso de portadores de carga. Deste modo, devido aos efeitos das fronteiras móveis, todos os condensadores e resistências na rede RC são elementos não lineares dependentes da largura da região mencionada.

A partir desta aproximação foram desenvolvidos modelos para o díodo (Massol 1993; Leturq1996b), BJT (Gillet 1995), GTO (Kallal, 1994) e para IGBT (Debrie 1996), todos implementados no simulador ESACAP™.

Este modelo, baseado nas séries de Fourier pode ser implementado nos simuladores como MATLAB/Simulink, usufruindo das vantagens das capacidades da matriz MATLAB na reprodução de um modelo compacto (Palmer, 2002). A Figura 2.8 ilustra o diagrama de bloco esboço do modelo no Simulink.

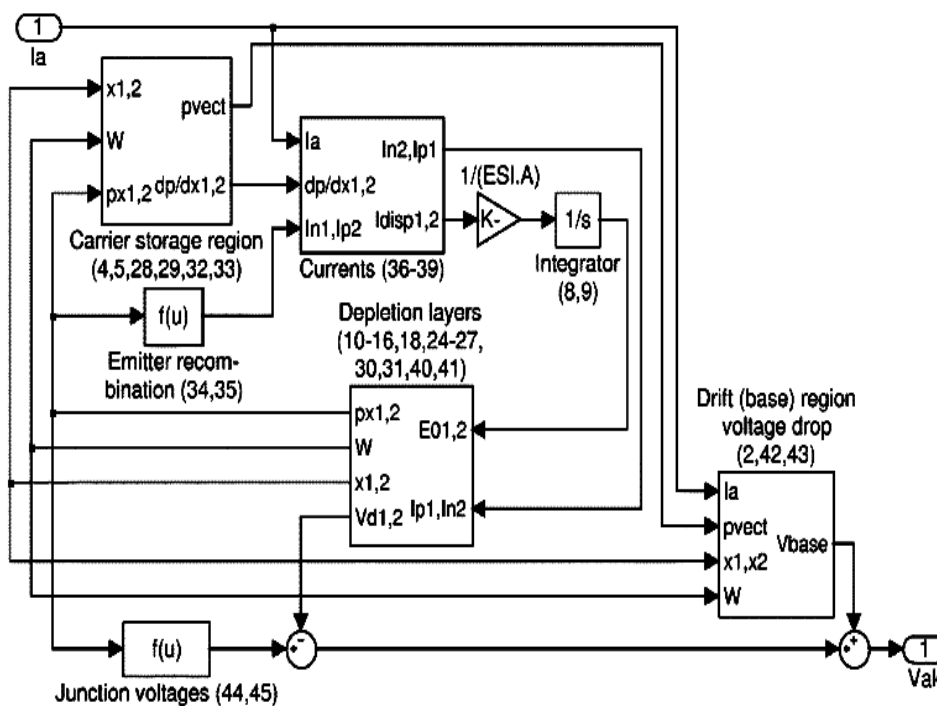


Figura 2. 8 - Diagrama de bloco do díodo no Simulink (Palmer et al, 2008)

Comparado com outros modelos analíticos, modelos baseado neste método fornece um melhor compromisso entre a velocidade de simulação e precisão (Kang, 2003).

2.2.2.3. MODELO DE CARGA DISTRIBUÍDA

A equação dos modelos de carga distribuída é baseada na física dos semicondutores, porém, o facto da inexistência de soluções exatas, a representação matemática é encontrada por aproximação da solução.

Alguns autores utilizam diferentes técnicas matemáticas para otimizar as escolhas de funções e obter uma aproximação fechada para equação de difusão e suas condições limites. Genericamente a aproximação para a distribuição $p(x, t)$ tem a seguinte forma:

$$p(x, t) = p_0(x, t) + \sum_i g_i(t) f_i(x) \quad (2.35)$$

onde $p_0(x, t)$ é a distribuição em equilíbrio, que é a solução da equação de difusão com $dp / dx = 0$. O somatório é a representação da dinâmica de cargas.

A variedade das funções é utilizada pelos termos $f_i(x)$: Trigonométrica, polinómio, exponencial, hiperbólico, etc. Na equação (2.35), $f_i(x)$ são funções experimentais e $g_i(x)$ são coeficientes dependentes de tempo e são determinados assim que uma boa aproximação é alcançada. Normalmente, poucas incógnitas são incluídos na soma da equação (2.35) com finalidade de limitar o esforço computacional. Algumas funções são soluções exatas particulares da equação de difusão se condições fronteiras específicas e condições iniciais são realizadas. Porém, apesar das restrições algumas vezes são utilizados para construção de solução geral.

Uma outra abordagem deste método é permuta de função de aproximação por tempo derivativo na equação de difusão. Na maioria dos casos, a soma de termos de produtos é também utilizado para separar variáveis

$$\frac{dp}{dt} = \sum_i h_i(t) f_i(x) \quad (2.36)$$

A solução (2.36) tem a forma de (2.35). Nos casos mais simples, apenas um termo $f_i(x)$, que tem na adição uma dependência linear x , é utilizado (Allen R.Hefner, 1990). Precisoões elevadas são alcançadas com mais termos $f_i(x)$ e dependências diferentes ou variáveis em x .

2.2.3. Modelos numéricos

Este modelo baseia-se nos métodos numéricos para descrever os semicondutores, utilizando o método dos elementos finitos ou métodos das diferenças finitas. A modelação numérica faz o uso de equações diferenciais parciais para descrever com elevada precisão os fenómenos físicos dentro de semicondutor, cifrando-se da deriva de portadores e componentes de difusão, geração de portadores e efeitos de recombinação, e entre carga espacial e campos elétricos.

Do ponto de vista de engenharia, o grau de precisão que é atingido por um modelo numérico completo nem sempre é necessário ou mesmo justificado, como por exemplo, se alguns dados de entrada, tal como o perfil de dopagem, é apenas conhecido com uma precisão limitada (Tan & Tseng, 1999). Nestes casos, modelos numéricos podem ser simplificados. Concentrações médias de dopantes são normalmente utilizadas. Em algumas circunstâncias, porém, o conhecimento sobre o perfil de dopagem é importante (Kraus & Mattausch, 1998).

Este modelo é muito usado na otimização por permitir observar os efeitos causados pelas variações dos componentes geométricos ou condições de operação, para além da sua capacidade na simulação de fenómenos 2D e 3D. Para os fabricantes de dispositivos que pretendem avaliar o desempenho dos seus dispositivos na aplicação de circuitos de eletrónica de potência, os métodos numéricos são uma abordagem adequada.

2.2.4. Modelos híbridos

Modelo híbrido, também conhecido como modelo semi-numérico é uma técnica baseada na combinação de modelo numérico e modelo analítico, e resulta de que certos fenómenos físicos nos dispositivos de potência são muito difíceis de simular realisticamente utilizando apenas equações analíticas, em particular, os efeitos de armazenamento de carga nas regiões fracamente dopadas. Isto, no entanto pode levar à falta de precisão dependendo dos pressupostos realizados e para evitar essas desvantagens um modelo híbrido é utilizado (Goebel H. , 1994).

Alguns artigos na literatura também têm feito referência sobre a causa do surgimento deste método. Por exemplo, os autores (Tan & Tseng, 1999, pp. 6-7) afirmam que, o longo tempo de computação necessário pelos modelos numéricos, também esteve na origem deste modelo.

A conceção fundamental do método é utilizar um algoritmo numérico rápido que resolve a equação de difusão ambipolar na zona n- do semiconductor, e usar modelos analíticos simples para modelar as restantes zonas do semiconductor. O trabalho realizado nesta tese tem como base o modelo híbrido, por isso nas próximas subsecções é dedicada uma atenção especial aos constituintes deste modelo no contexto de modelação de semicondutores.

2.2.4.1. MODELOS HÍBRIDOS: Método de diferenças finitas

O método de diferenças finitas foi desenvolvido por (Vogler & Schroder, 1992) para resolver a equação de difusão ambipolar. O modelo inclui a dependência espacial de distribuição de portadores de carga, recombinação de lacunas, efeitos de avalanche, dopagem e perfil de tempo de vida de portadores e efeitos de camadas de buffers. Desta forma, a utilização de parâmetros efetivos que não sejam físicos tais como concentração de portadores médios, mobilidade média, etc., são evitados.

Em contraste, nas soluções já com base proposta na literatura (H. Goebel, K. Hoffmann, 1992), (Vogler & Schroder, 1992) Métodos de Diferenças Finitos (MDF), a derivada temporal é também discretizada produzindo um sistema de equações algébricas, com claras desvantagens para implementação nos simuladores de circuito.

2.2.4.1.1. Discretização temporal

No domínio de tempo contínuo é subdividido em passos de tempo discreto..., t_{j-1} , t_j , t_{j+1} ..., a distância entre t_j e t_{j+1} é denotado Δt_j . Uma técnica de integração amplamente utilizada é a fórmula de diferenciação ou método de Euler. Existem duas direções de integração expressas por diferenças que podem ter a forma:

Forward Euler (FE) ou FDF1

$$\left. \frac{\partial p}{\partial t} \right|_{t_j} = \frac{p(t_{j+1}) - p(t_j)}{\Delta t_j} \quad (2.37)$$

Backward Euler (BE) ou BDF1

$$\left. \frac{\partial p}{\partial t} \right|_{t_j} = \frac{p(t_j) - p(t_{j-1})}{\Delta t_{j-1}} \quad (2.38)$$

Quando combinado, formam-se métodos Crank & Nicholson ou Regra Trapezoidal:

$$\frac{\partial p}{\partial t} = \frac{1}{2}(\text{FE}) \Big|_{t_j} + \frac{1}{2}(\text{BE}) \Big|_{t_{j+1}} \quad (2.39)$$

Ou geralmente

$$\left. \frac{\partial p}{\partial t} \right|_{t_j} = (1 - \gamma)(\text{FE}) \Big|_{t_{j+1}} \quad (2.40)$$

$\gamma = 0$: Forward Euler puro

$\gamma = \frac{1}{2}$: Crank & Nicholson

$\gamma = 1$: Backward Euler puro

Apesar de numericamente seja o método mais simples, o método FE inclui a desvantagem nas convergências lineares nos domínios de tempos e estabilidade condicional. No pior dos casos o método leva a erros não negligenciável depois de alguns poucos espaços de tempo.

O método BE fornece convergência linear no domínio de tempo e estabilidade incondicional. A solução é implícita.

O método TR fornece convergência quadrática no domínio de tempo e estabilidade incondicional. A grande desvantagem é toque que pode levar a instabilidades.

Melhores resultados seriam alcançados com $0.5 < \gamma \leq 1$

2.2.4.1.2. Discretização espacial

A partir de desenvolvimento da série de Taylor, considere-se as equações:

$$p(x_0 + h) = p(x_0) + h \frac{\partial p}{\partial x} + O(h^2) \quad (2.41)$$

$$p(x_0 - h) = p(x_0) - h \frac{\partial p}{\partial x} + O(h^2) \quad (2.42)$$

A derivada de ordem um pode ser aproximada por diferentes esquemas:

Diferenças progressivas ou diferenças a jusante a partir da equação (2.41):

$$\frac{\partial p}{\partial x} = \frac{p(x_0+h) - p(x_0)}{h} \quad (2.43)$$

Diferenças regressivas ou a montante partir da equação (2.42)

$$\frac{\partial p}{\partial x} = \frac{p(x_0) - p(x_0-h)}{h} \quad (2.44)$$

Diferenças centrais, subtrai-se (2.42) de (2.41)

$$\frac{\partial p}{\partial x} = \frac{p(x_0+h)-p(x_0-h)}{2h} \quad (2.45)$$

Para aproximação da 2ª derivada faz-se uma abordagem semelhante à anterior, através do polinómio de Taylor de segunda ordem:

$$p(x_0 + h) = p(x_0) + h \frac{\partial p}{\partial x} + \frac{h^2}{2} \frac{\partial^2 p}{\partial x^2} + O(h^3) \quad (2.46)$$

$$p(x_0 - h) = p(x_0) - h \frac{\partial p}{\partial x} + \frac{h^2}{2} \frac{\partial^2 p}{\partial x^2} + O(h^3) \quad (2.47)$$

Somando (2.46) e (2.47) obtém-se:

$$\frac{\partial^2 p}{\partial x^2} = \frac{p(x_0+h)-2p(x_0)+p(x_0-h)}{2h} \quad (2.48)$$

2.2.4.1.3. Modelação

Goebel e Hoffmann (1992, pp. 427-428) entendem que é utilizado o método de diferenças finitas para resolver a equação EDA. As condições limites são expressas pelas diferenças que podem ter na forma:

Relembrando a equação da EDA:

$$\frac{\partial p}{\partial t} = D \frac{\partial^2 p}{\partial x^2} - \frac{p}{\tau} \quad (2.49)$$

$$\left. \frac{\partial p}{\partial x} \right|_j = \frac{-p_{j+2} + 4p_{j+1} - 3p_j}{2h} \quad (2.50)$$

$$\left. \frac{\partial^2 p}{\partial x^2} \right|_j = \frac{p_{j+1} - 2p_j + p_{j-1}}{h^2} \quad (2.51)$$

onde o índice “j” indica o número de malhas de pontos. A equação 2.50 é um exemplo de diferenças progressivas para a derivada no espaço. Todavia, temos aqui outras vias para de representar as diferenças centrais de regressivas. O tempo pode ser também discretizado:

$$\left. \frac{\partial p}{\partial t} \right|_{k+1} = \frac{p_{k+1} - p_k}{h} \quad (2.52)$$

e resulta um sistema de equações algébricas.

Substituindo (2.51) e (2.52) em (2.49) obtém-se:

$$p_{j,k+1} = C_1(p_{j-1,k} + p_{j+1,k}) - C_2 p_{kj} \quad (2.53)$$

com

$$C_1 = \frac{\Delta t D}{h^2} \quad (2.54)$$

$$C_2 = \frac{2\Delta t D}{h^2} + \frac{\Delta t}{\tau} - 1 \quad (2.55)$$

A equação (2.53) dá a densidade de portadores $p_{j,k+1}$ explicitamente nos termos de três valores $p_{j-1,k}$, $p_{j,k}$ e $p_{j+1,k}$ de passo de tempo precedente. Por outro lado, a abordagem implícita não é necessária na resolução de sistema de equação para ter a distribuição de portadores de cargas. No entanto, há necessidade de defender a estabilidade e convergência da solução numérica do erro de truncatura (Lindsay, 2003):

$$2D\Delta t \leq h^2 \quad (2.56)$$

A equação (2.56) pode ser utilizada para determinar o número de pontos da malha (N). O valor máximo de passo de tempo é determinado por parâmetros físicos. O valor de Δt pode ser determinado em função de N:

$$\Delta t = \frac{L^2}{2DN^2} \quad (2.57)$$

Para um díodo de potência, o valor típico é aproximadamente 1ns. Se o valor do N escolhido for 50, adotando a constante de difusão $D = 2 \cdot 10^{-3} \text{m}^2 \text{s}^{-1}$ e o comprimento da região fracamente dopada $L = 100 \mu\text{m}$, os resultados possíveis do passo de tempo estão na

A diferença máxima entre dois passos de tempo é de 1ns, e é este valor que o algoritmo pode suportar sem se tornar instável.

A dimensão do intervalo Δt é calculada no simulador do circuito e caso ultrapasse o limite máximo a execução do algoritmo numérico é repetido (Goebel H. , 1994, p. 499).

A inconveniência do método explícito é que um passo de tempo pequeno deve ser utilizado para manter a instabilidade porém, computacionalmente é mais rápido e mais simples. A estabilidade e precisão dos híbridos também dependem da escolha de funções básicas de elementos.

Tabela 2. 1 - Valores máximos possíveis no cálculo de passo de tempo.

N	L (μm)	$\Delta t_{\text{máx}}$ (ns)
50	100	1
20	100	0.125
10	20	1

2.2.4.2. MODELOS HÍBRIDOS: Método de elementos finitos

Este método é especialmente adequado quando o problema é definido sobre o domínio espacial geometricamente complexo. Por esta razão, o método de elementos finitos tem sido aplicado com sucesso em grandes campos de problemas de engenharia tais como mecânicas dos materiais, dinâmica dos fluidos, transferência de calor (Quiza et al, 2012). Um método de elemento finito variacional de solução tipo foi introduzido inicialmente utilizando princípio de energia potencial mínimos e elementos triangular linear foi visto no campo de matemática aplicada (Courant, 1943).

Considere uma abordagem cuja função $p(x, t)$, delimitada pelo domínio Ω , é aproximada por um conjunto $N_i(x)$ definidas como subdomínio Ω_e , formado por uma malha de elementos finitos. As escolhas dessas funções, baseadas no conjunto de valores finitos r , determinarão os diferentes tipos de elementos finitos. As funções $N_i(x)$, são conhecidas por funções de base e a função $p(x, t)$ é relacionada por:

$$p(x, t) \cong \hat{p}(x, t) = \sum_{e=1}^n (\sum_{i=1}^r \alpha_i^e N_i^e) \quad (2.58)$$

$$\hat{p} = \alpha N$$

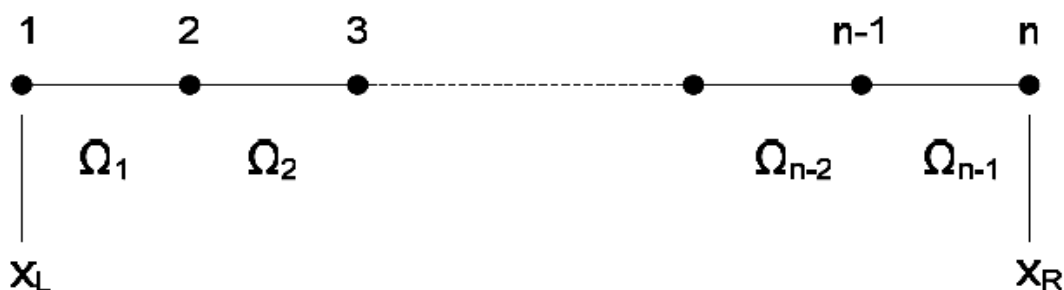


Figura 2. 9 - Nós e elementos finitos num domínio discretizado

Independentemente das abordagens utilizadas para a solução de um problema contínuo através dos métodos de elementos finitos, seguirá sempre um procedimento passo a passo de forma ordenadamente:

1. Discretização do domínio

Discretização do domínio dentro do elemento finito é o primeiro passo neste método. É o mesmo que substituir o domínio tendo um número finito de graus de liberdade por um sistema tendo número finito de graus de liberdade. A forma, o tamanho, números e configuração de elementos tem que ser escolhido corretamente para que o domínio seja simulado o mais próximo possível com o mínimo esforço computacional para alcançar a solução.

2. Funções de interpolação

A função interpolação representa a variação de campo acima dos elementos. Concetualmente, este é um passo central na FEM. Frequentemente, polinómios são selecionados como funções de interpolação para variável campo porque são diferentes e fáceis de integrar. A magnitude do variável campo bem como a magnitude das duas derivativas podem ser desconhecidos nos nós.

3. Encontrar equações elementares

Uma vez estabelecida a expressão do problema matematicamente, independentemente do método utilizado, pela formulação fraca da equação utilizada, as aproximações utilizadas na interpolação são substituídas na expressão preferida alcançando desta forma as equações elementares.

4. Sistema de equações global

A formação do sistema de equações obtém-se através da combinação de várias equações elementares utilizando regras de conectividade simples.

5. Aplicação das condições fronteiras

A solução não pode ser obtida sem aplicação das condições fronteiras. Elas refletem os valores conhecidos para determinados variáveis primárias. A imposição das condições limites modifica as equações globais.

6. Resolução de sistemas equações

6. Resolução de sistemas equações

Equações globais modificadas são resolvidas para incógnitas primárias nos nós. Podem ser utilizados métodos diretos e iterativos.

O MEF tem sido utilizado em grande escala para todos os problemas físicos que são descritos por equações diferenciais. Este método, de acordo com Chen (2005) apresenta várias vantagens que tem contribuído para o aumento da sua utilização, como:

- Pode manipular geometria irregular de modo mais conveniente.
- A largura dos elementos pode ser variada. Esta propriedade permite que os elementos tenham largura de acordo com gradiente da função objetivo
- Materiais não homogêneos podem ser modelados com facilidade.
- Maior precisão dos resultados comparado ao MDF.
- A estrutura bem definida e sua versatilidade tornam possível o desenvolvimento de software de utilização geral para aplicações.

Nos próximos parágrafos vamos detalhar e demonstrar como o método é derivado na forma exigida.

2.2.4.2.1. Método Rayleigh-Ritz aplicado a EDA

Este método desenvolvido por Ritz (1909), é um modelo analítico através do qual a minimização de um funcional, conhecida como função Π , que descreve a distância de um percurso limitado nas extremidades por uma função qualquer que descreve a via.

Em (Araújo, Chibante, & Carvalho, 2008) foi proposta uma nova abordagem para a resolução da EDA através de formulação variacional utilizando uma aproximação com elementos finitos unidimensionais.

Partindo da equação básica, a EDA:

$$\frac{\partial p}{\partial t} = D \frac{\partial^2 p}{\partial x^2} - \frac{p}{\tau} \quad (2.59)$$

e condições limite:

$$\frac{\partial p}{\partial x}(x_1, t) = \frac{I_n(x_1, t)}{2AqD_n} - \frac{I_p(x_1, t)}{2AqD_p} = f(t) \quad (2.60)$$

$$\frac{\partial p}{\partial x}(x_r, t) = \frac{I_n(x_r, t)}{2AqD_n} - \frac{I_p(x_r, t)}{2AqD_p} = g(t) \quad (2.61)$$

Araújo demonstrou que a EDA pode ser resolvida por uma formulação variacional de (2.59) a (2.61) por:

$$\Pi = \int_{\Omega} \left[\frac{1}{2} \left(\frac{\partial p}{\partial x} \right)^2 + \frac{p^2}{2D\tau} + \frac{p}{D} \frac{\partial p}{\partial t} \right] d\Omega - \int_s [(f(t) + g(t))p] ds \quad (2.62)$$

onde Ω e s , são o domínio do problema e a fronteira respetivamente. Aproximando $p(x, t)$ na forma de soma de funções elementares:

$$p(x, t) = \sum N_i(x)P_i(t) \quad (2.63)$$

Substituindo (2.63) na equação (2.62) e minimizando a funcional Π resulta num sistema de equações diferenciais ordinárias (EDOs) com a seguinte estrutura matricial:

$$[M] \left[\frac{\partial p}{\partial x} \right] + [G][p] + [F] = [0] \quad (2.64)$$

Os elementos finitos que pertencem as matrizes são dados por:

$$[M_e] = \int_{x_l}^{x_r} \frac{1}{D} N^T N dx .$$

$$[G_e] = - \int_{x_l}^{x_r} \frac{\partial N^T}{\partial x} \frac{\partial N}{\partial x} dx + \int_{x_l}^{x_r} \frac{N^T N}{D\tau} dx \quad (2.65)$$

$$F_e = -[(f + g)N^T]_{x_l}^{x_r}$$

A abordagem feita acima permite a analogia elétrica com a equação (2.64) e a solução de um nó composto por resistências, condensadores e, fontes de corrente (ver Figura 2.10), em que tensões em cada nó são uma imagem de concentração de cargas $p(x, t)$.

As equações do sistema definida em (2.64) são resolvidas num simulador de circuitos elétricos (IsSPICE) por meio da analogia com um sistema de equações com malhas RC:

$$[C] \left[\frac{\partial v(t)}{\partial t} \right] + [G][v(t)] + [I] = [0] \quad (2.66)$$

As propriedades dos dispositivos em cada elemento podem ser parâmetros de sub-circuitos que permitem soluções para materiais heterogéneos, o que facilita o aumento de precisão por adição de mais elementos para a solução.

A semelhança com uma rede elétrica é indubitavelmente uma das grandes vantagens do modelo já que a elevada precisão que é exigida dos modelos implica uma implementação nos simuladores que fornecem alguns processos de inserção do modelo no código do

simulador. Ainda, uma vantagem importante é que a implementação dos modelos baseados na física nos simuladores SPICE padrão é simples (Araújo, Chibante, & Carvalho, 2008, pp. 6-7).

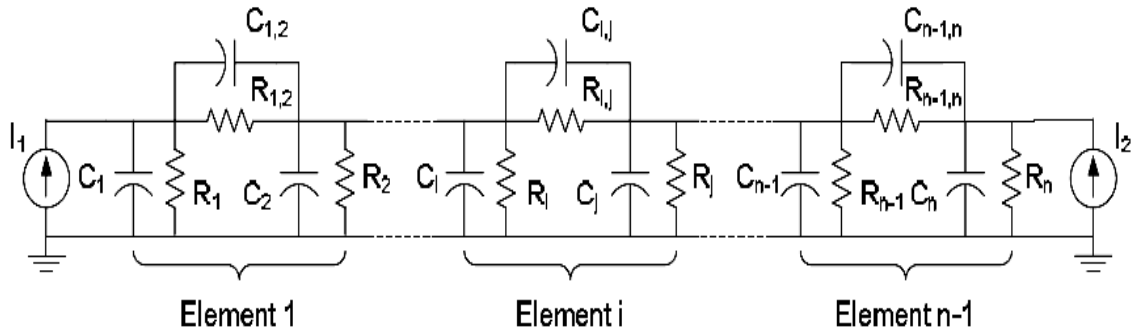


Figura 2. 10 - Rede RC com n-1 elementos finitos unidimensional

Por causa de alteração das condições fronteiras, as resistências e condensadores da rede RC são função de largura de base instantânea $W_B = x_r - x_l$. A largura dos elementos espaçado de forma igual os cálculos é feito por:

$$L_{Ee} = \frac{W_b}{n} \quad (2.67)$$

2.3. Algoritmos de otimização

Um dos princípios mais fundamentais no nosso universo é a procura de um estado ótimo (Weise, 2007). As técnicas de otimização focam sobretudo na procura de mínimo de uma função objetivo, também por vezes conhecida como função de custo.

Em matemática, resolver problemas de otimização significa encontrar a melhor solução de um conjunto de soluções. Esses problemas podem ser divididos em duas categorias consoante as suas variáveis sejam contínuas ou discretas. No caso mais simples, um problema de otimização baseia-se na maximização ou minimização de uma função real de variável real, pela pesquisa de várias soluções possíveis.

As técnicas de otimização podem ser usadas na identificação de parâmetros de modelos dos dispositivos semicondutores, isto porque, alguns modelos atuais desses dispositivos são muito complexos e com um elevado número de parâmetros (Josef Dobes, 2007).

Em otimização de dispositivos de potência, as funções objetivo possíveis passam pela inclusão de dissipação de potência mínima ou tempo de comutação mínima.

Assim, neste capítulo, apresentam-se abordagens diferentes de otimização e as suas classificações, fundamentos matemáticos e informação de base, evidenciando os pontos fracos e fortes de cada um dos métodos.

2.3.1. Técnicas de otimização

As técnicas de otimização podem ser baseadas em algoritmos determinísticos, cuja execução fornece sistematicamente a mesma solução através da pesquisa completa do espaço de soluções, ou heurísticas que podem fornecer, em tempo útil, soluções distintas através da pesquisa parcial do espaço de soluções.

2.3.1.1. Meta-heurísticas

Em ciências da computação, meta-heurística² designa-se por um método de otimização de problemas através da frequente tentativa de melhorar a solução candidata relativamente a uma dada medida de qualidade. Isto pode manipular uma solução única completa ou um conjunto de soluções em cada interação. As heurísticas subordinadas podem ser procedimentos de alto (ou baixo) nível ou uma simples pesquisa local, ou apenas um método de construção (Voß *et al.*, 1998).

Segundo Birattari (2001), uma meta-heurística será bem-sucedida num dado problema de otimização se puder fornecer um balanço entre a prospeção do conhecimento de pesquisa acumulada e exploração de espaço de pesquisa para identificar regiões com alta qualidade de soluções num problema específico, próximo de forma ideal.

Nesta secção vamos discutir as diferentes técnicas de otimização segundo critérios gerais que podem ser utilizados no contexto da identificação de parâmetros de modelos de semicondutores. O foco principal será dado aos métodos *Simulated Annealing (SA)* e *Particle Swarm Optimization (PSO)*, que são métodos utilizados no desenvolvimento desta tese.

² <http://en.wikipedia.org/wiki/Metaheuristic> [consultada em 2012-01-15]

2.3.1.1.1. Algoritmo Genético

Algoritmo Genético (AG) é um algoritmo de pesquisa que se baseia na combinação de seleção natural e genética, adequado para encontrar múltiplas soluções para problemas complexos (Mathew, 2005). Este algoritmo possui características diferentes relativamente a outras técnicas de pesquisa em diferentes aspetos. Muitos autores inspiraram-se na natureza para definir algoritmo genético.

Neste algoritmo, uma população de cromossomas é gerada no qual, a cada cromossoma é atribuído n valores para n variáveis envolvidas no problema. Os cromossomas passam por operações genéticas, seleções, cruzamentos e mutações de forma a melhorar a sua aptidão.

Basicamente, AG resume-se num processo que envolve conjunto de trabalho de indivíduos chamada população direcionado para funções objetivo. Uma vez definida a função objetivo, o AG procede para inicializar a população de soluções e melhorá-las através de aplicação repetitiva de operações genéticas descrita no parágrafo anterior.

2.3.1.1.2. Programação Evolucionária

Programação Evolucionária (PE) é uma estratégia de otimização estocástica semelhante a AG, que atribui ênfase na ligação comportamental entre pais e descendentes, em vez de estimular operadores genéticos específicos como observado na natureza.

Tanto a AG e PE, são métodos úteis de otimização quando outras técnicas tais como uma rampa descendente ou descoberta analítica direta não são possíveis. Função de real valor ou combinatorial nos quais a superfície de otimização ou *fitness* do espaço de pesquisa é irregular, possuindo muitas soluções ótimas locais, são adequados para a Programação Evolucionária.

2.3.1.1.3. Pesquisa Tabu

Pesquisa Tabu (PT), proposto por Glover (1977), é uma meta-heurística utilizada nos problemas de otimização combinatória.

O método PT utiliza uma estrutura de memória conhecida como Lista Tabu, que restringe os possíveis membros de vizinhanças para que a pesquisa possa prosseguir. Desta forma, encontrado um ótimo local, a pesquisa não será capaz de visitar a zona de espaço de solução. A memória preserva um número de estados visitados anteriormente junto com um número de estados que podem ser considerados indesejados. A definição de um estado, a

área a volta e o comprimento da lista tabu são parâmetros críticos do projeto. Entre os parâmetros de Tabu, os dois mais utilizados são: Aspiração e Diversificação.

Na PT, um número máximo de iterações é permitido na execução para que computacionalmente se torne prático. Uma descrição mais detalhada sobre PT pode ser encontrada em Glover e Laguna (1999).

2.3.1.1.4. Hill Climbing (Subida de Montanha)

Hill Climbing (HC) é método de algoritmo de otimização muito antigo e simples para uma única função objetivo (Russell & Norvig, 2002). Normalmente utiliza uma operação de pesquisa de parâmetros a fim de gerar candidatos de primeira solução e, a partir disso procede a operação unária cuja finalidade é produzir descendente.

Esta técnica de otimização é boa para encontrar um ótimo local porém, não é garantido encontrar melhor solução possível (ótimo global) entre todas as soluções possíveis (espaço de pesquisa). Assim sendo, podemos dizer que o maior problema do HC reside na sua convergência prematura, isto é, fica facilmente preso num ótimo local. Em contrapartida, a sua vantagem maior é a sua simplicidade, a dificuldade do núcleo geralmente sendo o modelo da função vizinhança.

2.3.1.1.5. Colónia de Formigas

Otimização Colónia de Formigas (OCF) tem sido formalizada dentro de meta-heurística para problemas de otimização combinatória por Dorigo et al. (1999). OCF foi inspirado no comportamento irrequieto de algumas espécies de formigas. Estas formigas colocam feromonas no chão com objetivo de marcar alguns caminhos favoráveis que devem ser seguidos por restantes membros da colónia. Portanto, a OCF explora um mecanismo similar para resolver os problemas de otimização.

A aplicação deste método é vasta, principalmente nos problemas de otimização discreta para que alguns mecanismos de construção de solução possam ser concebidos.

A característica de algoritmos OCF é seu uso explícita de elementos de soluções anteriores. De facto, eles conduzem uma solução de baixo nível construtivo, como Grasp (Feo & Resende, 1995) faz, mas incluindo-o num Framework e randomizando a construção através de Monte Carlo.

2.3.1.1.6. Simulated annealing (Recozimento Simulado)

Baseado na teoria de Metropolis, Kirkpatrick et al. (1983) inspirou para desenvolver o algoritmo Simulated Annealing (SA) no processo de otimização global, no início da década de oitenta e aplicou em vários problemas de otimização combinatória. Autonomamente, Vlado Černý deduziu uma abordagem semelhante no problema “caixeiro-viajante” (Lawler, Lenstra, Kan, & Shmoys, 1985).

Portanto, podemos afirmar que o nome surgiu da sua analogia com o processo de recozimento físico dos sólidos, em que um metal no estado sólido é aquecido a uma temperatura crítica e posteriormente deixado ao arrefecimento muito lento até alcançar a sua configuração mais regularmente possível na rede cristalina, isto é, o estado de energia mais baixo, tal que os defeitos dos cristais desapareçam (Henderson, Jacobson, & Johnson, 2003). A taxa de arrefecimento tem que ser suficientemente lenta para que a configuração resultante seja um sólido com integridade estrutural superior.

Em cada iteração de SA aplicado para o problema de otimização discreto, é calculada a função objetivo para duas soluções (melhor solução atual e nova solução selecionada) que são comparadas.

A característica chave do algoritmo SA é que fornece uma via para escapar a mínimos locais pela permissão do movimento de “subida” (i.e., movimentos que pioram o valor da função objetivo).

Atualmente muitos são os artigos disponibilizados em linha (em World Wide Web) que fornecem uma boa visão de desenvolvimento teórico de Simulated Annealing e domínios de aplicações, onde surgem artigos como Eglese (1990), Fleischer (1995), Koulamas et al (1994), Romeo e Sangiovanni-Vicentelli (1991) e livros que baseiam-se inteiramente neste algoritmo como é o caso dos autores Aarts e Korst (1989), Van Laarhoven e Arts (1988) e mais recentemente Chibante (2010).

Descrição do algoritmo

Para detalhar melhor o funcionamento de algoritmo utilizado em parte no desenvolvimento desta tese, vamos basear na teoria apresentada no livro de Chibante (2010), no qual menciona que nas temperaturas muito elevadas, os átomos das matérias sólidos, apresentam um nível de energia também elevada. O que acontece nas altas temperaturas é que as moléculas dos líquidos movem-se livremente em relação aos outros. Se os líquidos são arrefecidos lentamente, a mobilidade térmica é perdida.

Como a temperatura é reduzida gradualmente, a energia dos átomos diminui até um estado de energia mínima e então permanece praticamente fixo no sítio da rede substituída e, forma cristal.

A sua finalidade é fornecer uma boa aproximação para um mínimo global. A tarefa principal deste algoritmo é minimizar a energia interna do sistema, que é representado por uma função energia E (estado) que é uma função do estado atual. Quanto mais baixo for o valor da função de energia, mais próximo do estado de menor energia está e, conseqüentemente, próximo de uma solução ótima global. Uma heurística é utilizada para encontrar os vizinhos para estado atual. Após encontrar os vizinhos, a diferença de energia $\Delta E = E_2$ (estado2) – E_1 (estado1) são calculados para cada um dos novos estados propostos e são passados para uma função de probabilidade que determina o estado sucedido. No entanto, há também probabilidade de não ocorrer nenhuma transição e o sistema permanecer no estado atual. A probabilidade de transição de um estado para outro é uma função da diferença de função de energia em simultâneo com o parâmetro global chamado temperatura. O parâmetro da temperatura $T(t)$ é uma função do sistema de recozimento de tempo t e controle da capacidade de recozimento e uma vez que alcance o zero, o critério de paragem for considerada e algoritmo termina. A temperatura do sistema é muitas vezes representada como uma função exponencial $e^{-\frac{\Delta E}{T}}$ que diminui rapidamente nos estágios iniciais da computação e lentamente na parte final.

No caso da função de minimização, vamos supor que o estado atual encontra-se no ponto X_i com valores correspondentes da função objetivo dada por $f_i = f(X_i)$. De forma similar, o estado da energia de um sistema termodinâmico, de energia E_i no estado X_i é dado por:

$$E_i = f_i = f(X_i) \quad (2.68)$$

De acordo com critério de Metropolis, a probabilidade de alcançar o próximo estado, X_{i+1} depende da diferença nos estados de energia ou valores de funções nos dois pontos (estados) representado por:

$$\Delta E = E_{i+1} - E_i = \Delta f = f_{i+1} - f_i \quad (2.69)$$

e o novo estado, X_{i+1} pode ser encontrado utilizando a distribuição de probabilidade de Boltzman:

$$P[E_{i+1}] = \min \{1, e^{-\Delta E/kT}\} \quad (2.70)$$

é importante observar que se $\Delta E \leq 0$, a equação 2.70 resulta $P[E_{i+1}] = 1$ e portanto o estado X_{i+1} é sempre aceite. Isto é, a escolha lógica no contexto de minimização de função porque o valor da função em X_{i+1} , f_{i+1} é melhor (mais pequeno) que em X_i , f_i e por isso, o próximo estado deve ser aceite. Em vez disso, quando $\Delta E \geq 0$ o valor da função f_{i+1} no ponto (estado) X_{i+1} é pior (maior) do que se encontra no estado X_i . Segundo a maioria dos procedimentos de otimização convencionais, o estado X_{i+1} não pode ser aceite como o próximo estado no processo iterativo.

Contudo, a probabilidade do próximo estado (X_{i+1}) ser aceite, mesmo tendo pior valor da função objetivo em relação ao estado anterior (X_i), é definido segundo critério de Metropolis (Eq. 2.70).

Uma das especificidades do algoritmo recozimento simulado é a permissão de transições para estados onde a função ΔE seria positiva. Se a diferença da função de energia for positiva significa que o estado sucedido está mais longe da solução (pior) que o atual. Isto é realizado através da atribuição de probabilidades diferentes de zero a partir do estado atual para o pior estado. Assim, no algoritmo de recozimento é possível evitar ficar preso num mínimo local. O recozimento simulado ostenta algumas vantagens como, a capacidade para evitar a prisão nos mínimos locais e, ainda uma convergência ao executar número suficientemente grande (indefinido) de interações. Por outro lado, a principal desvantagem, que são comuns nos algoritmos de pesquisa local estocástica, prende-se com o facto de ter que confrontar com a dificuldade determinar o diagrama de arrefecimento, isto é, decidir a quantidade de interações suficiente em cada temperatura e, determinar a temperatura inicial. No que diz respeito a esta última, se iniciar com um valor muito alto, perde-se no tempo de computação, se for um valor alto, diminui a quantidade de pesquisa. A solução passa por ajustar o algoritmo para maximizar o seu desempenho. O algoritmo de recozimento está representado pelo seu fluxograma Figura 2.11.

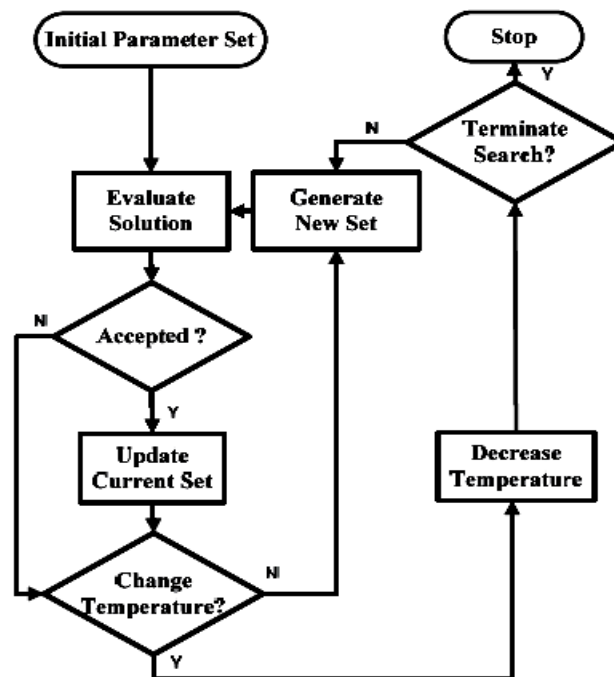


Figura 2. 11 - Fluxograma de algoritmo Recozimento Simulado

Todos os pré-requisitos mencionados influenciam significativamente o progresso do algoritmo e a escolha dos parâmetros e funções é, infelizmente, única para cada problema, o que torna difícil adaptar o algoritmo a um determinado problema.

Diagrama de arrefecimento

O ingrediente essencial para um ótimo desempenho do procedimento do algoritmo SA, é sem dúvida alguma, a definição de escala de arrefecimento $T(t)$, onde T é a temperatura e t é a medida do tempo de cálculo. Com ajuda da teoria do processo de Markov tem sido possível mostrar que diagrama de escala faz com que *Simulated Annealing* encontra a solução ótima de um dado problema (finito) no tempo infinito. Esses diagramas têm a forma (Hartmann & Rieger, 2002b):

$$T(t) = \frac{a}{b + \log(t)} \quad (2.71)$$

onde a e b são constantes positivas que dependem do problema. O esquema linear (ver Figura 2.12a).

$$T(t) = a - bt \quad (2.72)$$

aqui, a é a temperatura inicial e b é o tamanho do passo para diminuir a temperatura (normalmente $0.01 \leq b \leq 0.2$) e o esquema exponencial:

$$T(t) = ab^t \quad (2.73)$$

onde outra vez a é a temperatura inicial e b a taxa de arrefecimento (normalmente $0.8 \leq b \leq 0.999$).

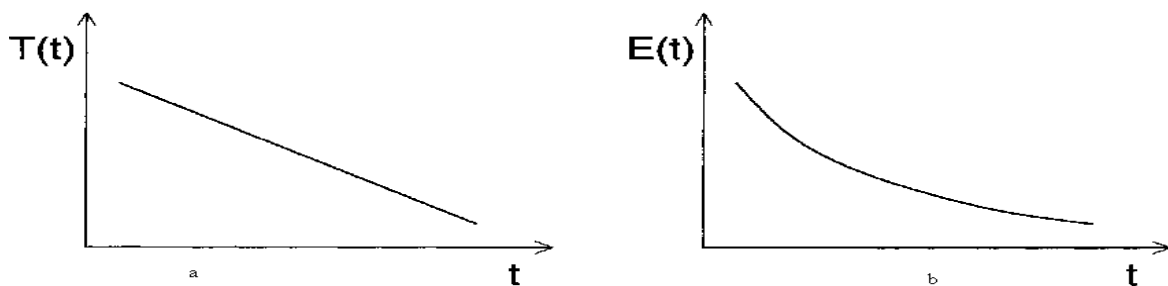


Figura 2. 12 - Simulated annealing: a - Escala de arrefecimento linear $T(t)$ e b – Esboço de de energia média resultante $E(t)$ como uma função de tempo.

Função objetivo (FO)

A função objetivo representa uma medida quantitativa da “bondade” de alguns sistemas complexos. Esta função depende da configuração detalhada de muitas partes do sistema, em que a finalidade é maximizar ou minimizar os valores de parâmetros de algumas propriedades (Kirkpatrick et al., 1983a). Para o problema em causa nesta tese a fórmula base da função objetivo, centra-se na definição de diferenças entre os resultados obtidos

experimentalmente com resultados simulados. A função objetivo pode ser expressa pela fórmula:

$$f_{obj} = \sqrt{\sum_c \sum_i \left(\frac{E_s(x_i) - E_e(x_i)}{E_e(x_i)} \right)^2} \quad (2.74)$$

onde $E_e(x_i)$ é o valor de dados experimental, $E_s(x_i)$ é o valor simulado e c representa a quantidade das curvas a ser otimizado.

Temperatura inicial

Este parâmetro, temperatura, deve ser bem definido uma vez que controla a regra de aceitação definido pela equação de distribuição de Boltzman. Não deve ser definido nem muito grande nem muito pequeno. Deve ser grande, o suficiente para permitir o algoritmo não fique preso no mínimo local e pequeno o suficiente, para não mover do ótimo global. A definição do seu valor depende da aplicação envolvida, uma vez que é relacionado com a grandeza dos valores de função objetivos. A diminuição do parâmetro da temperatura baseia-se numa escala de arrefecimento como o algoritmo converge para uma solução ótima (El-Naggar, et. Al., 2011).

Estimativa inicial

A definição da estimativa inicial ou população inicial é o requisito necessário na utilização dos métodos baseados em técnicas iterativas. Quanto mais próximo do ótimo for a estimativa inicial mais rápido será o processamento de um ótimo global.

Mecanismo de perturbação

Este parâmetro desempenha um papel importante na geração de novas soluções a partir de soluções atuais. Vamos considerar os pontos do espaço total representado por um vetor $s = (x_1, \dots, x_n)$. A geração da nova solução passa por utilização de um vetor $\sigma = (\sigma_1, \dots, \sigma_n)$ correspondente ao desvio-padrão associado a cada variável (parâmetro)

$$x_{i+1} = x_i + N(0, \sigma_i) \quad (2.75)$$

onde $N(0, \sigma_i)$ é um número aleatório Gaussiano com média zero e desvio padrão σ_i .

Crítério de paragem

São varias as técnicas para controlar o fim do algoritmo. Alguns exemplos estão citados em baixo:

- definição de número máximo de iterações
- definição do valor mínimo da função objetivo
- valor mínimo da taxa de aceitação
- valor mínimo da temperatura inicial

2.3.1.1.7. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (Otimização de Enxame de Partícula) é um método de otimização heurística global originalmente proposto por Kennedy e Eberhart (1995). Mais tarde publicaram um livro (James F. Kennedy, 2001) e vários artigos sobre PSO (Eberhart & Shi, 1998), (Kennedy & Eberhart, 1995b) (Eberhart & Shi, 1998b) (Eberhart & Shi, 2001), num dos quais fez um estudo sobre o desempenho através de quatro funções não-lineares e que posteriormente foi adotada por vários investigadores nesta área (Kennedy & Eberhart, 1995c).

Para perceber melhor o funcionamento do PSO ou enxames é necessário entender-se o conceito de inteligência de enxame (*Swarm intelligence*), no qual sobressai a inteligência do coletivo. Portanto, apenas para dizer que swarm intelligence (inteligência do grupo) é o termo utilizado para designar sistemas de inteligência artificial onde o comportamento coletivo dos indivíduos numa população causa soluções simples e coerentes. A inteligência coletiva é uma característica de sistemas constituídos por “agentes” (partículas) pouco inteligentes cuja capacidade individual é limitada mas capazes de apresentar comportamentos coletivos. Tais comportamentos apresentam as seguintes propriedades:

Proximidade: as partículas devem ser capazes de interagir;

Qualidade: essas partículas devem ser capazes de avaliar os seus comportamentos;

Diversidade: O sistema reage em situações inesperadas;

Estabilidade: devem resistir a variações ambientais que afetam o comportamento da partícula;

Adaptabilidade: capacidade para adaptar as variações ambientais;

Por outro lado o algoritmo baseia-se na metáfora de interação social, pesquisa um espaço por ajustamento de trajetórias de vetores individuais, chamados de partículas que são conceituados como pontos que se deslocam no espaço multidimensional. As partículas individuais são deslocadas estocasticamente para as posições dos seus próprios melhores

desempenhos anteriores e os melhores desempenhos anteriores dos seus vizinhos (Clerc & Kennedy, 2002). Cada partícula “voa” na direção que se baseia a sua experiência e na experiência do modelo global. Partículas individuais movem estocasticamente para a posição em função da velocidade presente, melhor desempenho anterior e melhor desempenho anterior do grupo (Lazinica, 2009).

O exemplo da metáfora utilizada para comparar este método baseia-se na comparação feita com bandos de pássaros. Quando os pássaros se movimentam de um lugar para outro é similar ao desenvolvimento de solução de enxame, boa informação é igual a solução mais otimista, e o recurso alimentar é igual à solução mais otimista durante todo o percurso. A solução mais otimista pode ser trabalhada em algoritmo de otimização de enxames de partículas por cooperação individual (Bai & Tongliao, 2010).

A simplicidade e facilidade de implementação constituem as principais vantagens deste algoritmo, por isso também, é amplamente utilizado nos campos de otimização, classificação dos modelos, estudo de máquinas, treino de redes neuronais, processamento de sinal, controlo de sistemas difusos, controlo de adaptação automática, etc. (Zheng Jianchao, 2004).

Algoritmo básico de Particle Swarm Optimization (PSO)

Kennedy e Eberhart (1995b) compararam o comportamento dos enxames na natureza tais como os pássaros, peixes, abelhas, etc. com as partículas constituintes no desenvolvimento do algoritmo PSO. Este algoritmo combina experiências próprias com as experienciais coletivas em que a solução candidata é apresentada como partículas.

Num sistema PSO, cada partícula tendo duas propriedades de posição e velocidade representa uma solução candidata expressa pela função objetivo. Na iteração, a função objetivo é calculada para estabelecer o valor de custo de cada partícula utilizando a posição como entrada. A função custo determina qual posição é a melhor através de avaliação do desempenho das partículas. As partículas, individualmente “voam” no espaço de pesquisa com uma velocidade instantânea, ajustada consoante a sua melhor experiência e a melhor experiência do enxame. Por outras palavras, cada partícula utilizará tanto a informação da sua melhor posição (personal best) e as informações da melhor posição global (global best) (FANG, CHEN, & LIU, 2007).

Considerando um problema de otimização de D variáveis, um enxame de N partículas é inicializado em que a cada partícula é atribuída uma posição aleatória num espaço grande de dimensão D tal que cada posição de partícula corresponde a uma solução para o

problema de otimização. Assumimos que x representa a posição da partícula e v a velocidade de deslocamento da partícula. Então, todas as partículas que “sobrevivem” o espaço de dimensão D , estão sujeitos às regras de atualizações para novas posições, até que a posição de um ótimo global seja encontrado.

O processo (original) de implementação da versão global de PSO é como se segue (Shi & Eberart, 2001):

- 1) Inicializa uma população (vetor) de partículas com posições aleatórias e velocidades em D dimensões no espaço do problema.
- 2) Para cada partícula, calcula a função de custo (fitness) de otimização desejada nas D variáveis.
- 3) Compara função de custo da partícula com p_{best} da partícula. Se o valor atual é melhor que p_{best} então define o valor de p_{best} igual ao valor atual para a localização atual no espaço D -dimensão.
- 4) Compara a melhor função de custo da posição atual com a melhor posição do enxame anterior. Se o valor atual é melhor que g_{best} então redefine g_{best} para o vetor da partícula atual com índice e valor.
- 5) Altera a velocidade e posição da partícula de acordo com as equações (2.76) e (2.77), respetivamente:

$$v_{i+1} = wv_i + c_1 rand() \times (p_{i_{best}} - x_i) + c_2 rand() \times (g_{i_{best}} - x_i) \quad (2.76)$$

$$x_{i+1} = x_i + v_{i+1} \quad (2.77)$$

- 6) Loop para o passo 2) até um critério ser encontrado, normalmente um ótimo suficientemente bom ou número máximo de iterações.

onde w é o peso da inércia, c_1 e c_2 são duas constantes positivas e $rand()$ é um número aleatório entre 0 e 1.

Para alcançar um equilíbrio entre exploração global e local para acelerar a convergência para um verdadeiro ótimo, o valor peso da inércia (w), diminui linearmente com o número de iteração que tem sido utilizado:

$$w = w_{max} - \left(\frac{w_{max} - w_{min}}{n} \right) \times i \quad (2.78)$$

onde n é número da iteração, w_{min} e w_{max} são os valores iniciais e finais do peso de inércia, respetivamente, utilizado no PSO. O valores de $w_{max} = 0,9$ e $w_{min} = 0,4$ são recomendados na literatura.

As velocidades das partículas são limitadas à velocidade máxima V_{max} , que é um parâmetro importante. Determina a resolução, ou fitness, com que a região entre a posição atual e a posição alvo (a melhor até então) é pesquisada. Se V_{max} , for demasiado alta, as partículas podem ultrapassar os melhores resultados. Em contrapartida, se for demasiado pequeno, as partículas podem não explorar suficientemente para além das regiões localmente boas. De facto podem ficar presas no ótimo local, tornando incapaz de mover longe o bastante para atingir um posição melhor no espaço de problema.

As constantes c_1 e c_2 alteram os seus valores das “tensões” no sistema. Valores baixos permitem partículas vaguear longe da região alvo antes de ser arrastado de volta, enquanto os valores altos resultam num movimento bruto para, ou passado da região alvo.

Seleção de Parâmetros e Análise de trajetória

Vamos ainda apresentar neste algoritmo duas informações consideradas importantes no desempenho do algoritmo PSO. A primeira está relacionada com a seleção de parâmetros e a segunda é a forma como se realiza a análise de trajetória das partículas no espaço de pesquisa. Começamos por este último, a análise de trajetória é importante porque pode ajudar a determinar onde se encontra a posição de cada partícula em cada passo de evolução, e conseqüentemente pode auxiliar a clarificar o mecanismo de execução do algoritmo PSO assim como decifrar o porquê e quando o algoritmo pode ser bem-sucedido para resolver os problemas de otimização. Por outro lado, a seleção de parâmetros no algoritmo PSO tem a sua relevância visto que este algoritmo é sensível aos parâmetros escolhidos. A seleção de parâmetros no PSO é bastante importante e tem chamado atenção de muitos investigadores. Diferentes investigadores têm proposto diferentes conjuntos dos parâmetros tais como: $W=0.729$, $C_1=C_2=1.494$ (Clerc & Kennedy, 2002); $W=0.6$, $C_1=C_2=1.7$ (Trelea, 2003b); $W=0.729$, $C_1=2.041$, $C_2=0,948$ (Carlisle & Dozier, 2001).

Neste trabalho resolveu-se adotar o conjunto $W = [0,9 - 0,4]$ $C1=C2=2$, pelo facto de este conjunto apresentar maior balanço entre exploração³ e prospeção⁴, que um algoritmo deve manter (Holland, 1975).

Outro componente importante e que influencia o desempenho do algoritmo PSO é a estrutura ou topologia de comunicação das partículas. Ela é responsável pela forma como as partículas no enxame partilham informações e se comunicam. A procura de soluções ótimas pode priorizar tanto a velocidade de convergência, a qualidade de solução ou ambas. As principais topologias são: topologia local e topologia global representada na figura abaixo.

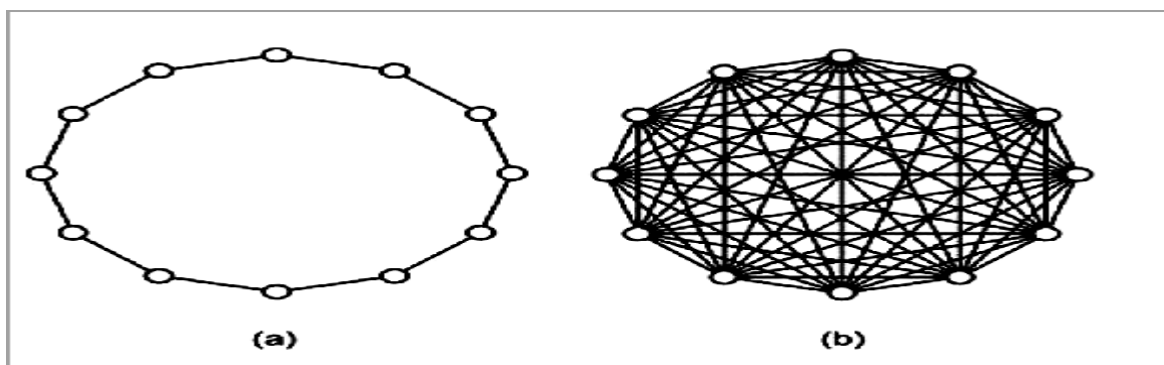


Figura 2. 13 - Topologia de comunicação das partículas: a) Local e b) Global

Na topologia local, ilustrado na Figura 2.13a, cada partícula possui dois vizinhos interligados em forma de anel e trocam informação apenas com os vizinhos de forma direta. Embora a iteração seja feita de forma mais lenta entre as partículas, este formato fornece uma melhor qualidade de soluções para problemas multimodais quando comparado com o mesmo fornecido pela topologia global.

Na topologia global (Figura 2.13b), o enxame está organizado em rede estrela e as todas as partículas estão ligados entre si. Esta organização utiliza o mecanismo de vizinhança global, denominado g_{best} na permutação de informação. Esta topologia permite uma convergência mais rápida comparada com a topologia anterior, uma vez que a informação da melhor posição é divulgada rapidamente entre todas as partículas, porém não garante a

³ Exploração é a tendência do algoritmo explorar novas regiões do espaço de pesquisa

⁴ Prospeção é a tendência para pesquisar uma região menor mais profundo.

qualidade da solução alcançada. Neste caso, o algoritmo pode atingir um mínimo local devido à sua convergência precoce.

Em (Clerc, 2006), o autor aponta para aquilo que considera os dois erros comuns existentes na equação do movimento (2.76) que às vezes são escritas na forma vetorial. O primeiro erro apontado pelo autor está relacionado com a definição de multiplicação de um vetor por um coeficiente, significando que todos os componentes, por exemplo o vetor $p_{best} - x_i$, são multiplicado pelo mesmo numero aleatório. Considera que isto é um erro no sentido que corresponde a um algoritmo diferente daquele do PSO mas também que esta fórmula pode ser considerada como alternativa.

Outro erro, ou alternativa, consiste da realização de uma factorização na primeira equação do movimento:

$$v_i = c_1 v_i + rand(0, c_{max})(p_{i_{best}} + g_{i_{best}} + 2x_i) \quad (2.79)$$

Nesta forma vimos que a próxima posição será então escolhido aleatoriamente de acordo com uma distribuição uniforme num híper paralelepípedo cuja aresta de dimensão D é o comprimento $C_{max}|p_d + g_d|$ e o centro é encontrado somando o vetor x, o vetor $c_1 v + c_{max} (p + g)/2$. Na verdade, poderia simplesmente descrever isso como alternativa ao

invés de um erro, porque esta distribuição é quase tão rica como a original.

Para uma compreensão mais profunda recomenda-se a consulta de alguns livros com informação significativa sobre PSO como: (Nedjah & Mourelle, 2006), (James F. Kennedy, 2001), (Clerc, Particle Swarm Optimization, 2006) (a & Tiwari, 2007).

2.3.2. Análise qualitativa dos algoritmos

Depois de descrever vários métodos de otimização, neste ponto comparam-se as diferentes técnicas, com foco principal no método PSO e SA, que constituem o alicerce deste trabalho. Relativamente às características analisadas, os seguintes estudos foram realizadas ao longo da secção 2.3:

- Eficiência;
- Eficácia;

- Parâmetros de entrada;
- Aplicações;
- Confiabilidade;
- Desenvolvimento;

2.3.2.1. Particle Swarm Optimization

PSO foi o método com maior quota na elaboração do estado de arte da teoria de métodos de otimização pelo facto de ser um dos métodos utilizado neste trabalho. O PSO é um método direcionado para realização de projetos de natureza complexa, com algumas semelhanças ao algoritmo genético (AG) mas com a capacidade de produzir resultados (convergência) de forma mais rápida, embora a precisão seja inferior ao AG. Diversas questões chave sobre PSO e AG têm sido apontadas por Rahamat-Samii (2003). O pouco ajuste de parâmetros que é necessário fazer é umas das razões que torna o PSO ser atrativo.

Ainda comparado com outros algoritmos, o método apresenta como principais vantagens a sua simplicidade e facilidade de implementação (Zheng Jianchao, Jie Jing, Cui Zhihua, 2004).

O algoritmo tem sido aplicado com sucesso num domínio amplo nas áreas investigação tais como treino de rede neural artificial (Eberhart and Shi 1998, Messerschmitt and Engelbrecht 2004), o problema fluxo de potência ótima (FPO) (Abido 2002), os problemas de atribuição das tarefas (Seleman et al. 2002), o problema de compromisso de unidade (Ting et al. 2003), modelo alinhamento de sequência múltipla (ASM) (Rasmussen and Krink 2004), otimização multiobjectivo (Coelho et al. 2004), otimização eletromagnética (Robinson & Rahmat-Samii 2003, Boeringer & Werner 2004), descoberta de tema proteína (Chang et al. 2004), etc.

O algoritmo apresenta algumas limitações comuns a outros algoritmos evolucionários. No campo de engenharia, por exemplo, é considerado um método muito centrado em problemas de otimização sem restrição. Para superar essas limitações muitos investigadores têm propostos numerosas variantes de PSO. Num estudo elaborado em (Hu, Eberhart, & Shi, 2003), os autores apresentaram um artigo sobre algoritmo PSO aplicado

nos problemas de otimização em engenharia onde pode se concluir que comparado com outros métodos, o PSO tem as seguintes vantagens:

- Mais robusto: Adequado para lidar com projetos não-lineares e não convexos em descontinuidade;
- Mais fácil: Não necessita de conhecimento no domínio específico, transformação interna das variáveis ou manipulações para lidar com restrições;
- Mais eficiente: Requer um número menor de avaliações de funções e com melhor ou a mesma qualidade dos resultados;

2.3.2.2. Simulated annealing

Simulated Annealing (SA) é uma técnica que tem atraído atenção para problemas de otimização de larga escala, especialmente aqueles onde um extremo global desejado está escondido entre muitos extremos locais.

Fleischer e Jacobson (1999) observaram que a configuração de espaço criado por uma instância de um problema de otimização discreta determina a eficiência de SA quando aplicado no problema. O valor principal da utilização do SA, portanto pode ser para execuções de tempo finito que obtém soluções quase ótimas razoavelmente rápidas. Sugere que deve-se focalizar no comportamento do tempo finito de SA em vez de resultados de convergência assintótica que dominam a literatura (Henderson, Jacobson, & Johnson, 2003).

O algoritmo SA tem provado ser uma boa técnica para resolver problemas de otimização discreta difíceis. Em engenharia de otimização SA tem emergido como uma ferramenta alternativa direcionado para problemas que são difíceis resolver por técnicas convencionais de programação matemática. A maior desvantagem apontada é que resolver um problema complexo pode ser extremamente lento, embora o processo seja convergente, utilizando muito mais tempo de processamento que alguns algoritmos convencionais.

Koulamas et al. (1994) num estudo focado especialmente em SA na aplicação de gestão de operações/produções e operações de pesquisa, concluíram que SA é uma ferramenta efetiva para resolução de muitos problemas em operações de pesquisa e que o grau de precisão que o algoritmo alcança pode ser controlado em termos de número de iterações e

funções da vizinhança, combinada com aumento de número de pesquisas a cada iteração pode resultar em soluções com maior probabilidade de convergir para ótimo global.

2.3.2.3. Algoritmo Genético

Algoritmo Genético (AG) é inspirado pela capacidade de aperfeiçoamento de sistemas biológicos através da sua evolução. Este algoritmo representa uma exploração inteligente de uma pesquisa aleatória dentro de um espaço de pesquisa para resolver um problema. Hartmann & Rieger (2002b) provaram que o algoritmo é muito mais eficaz do que uma pesquisa aleatória. Além disso, o algoritmo não é sensível ao ruído.

Num artigo apresentado por (Elbeltagia, Hegazyb, & Griersonb, 2005) que compara a formulação e resultados dos cinco últimos recentes algoritmos evolucionários, incluindo o AG e PSO. O método PSO foi geralmente encontrado com melhor desempenho que os outros métodos em termos de taxa sucesso e qualidade de solução, devido ao efeito de inércia e, surpreendentemente, o AG apresentou pior desempenho que os outros quatro algoritmos. Whitley et al. (1995), reportou no artigo que foi encontrado o pior desempenho do AG dos quatro.

Algoritmo Genético tem sido utilizado em vários problemas de otimização e pesquisa (Back Fogel and Michalewicz, 1997; Bramlette and Bouchard, 1991; Callahan and Weeks, 1992; Gen and Chang, 1997; Lucasius and Kateman, 1989).

2.3.2.4. Colônia de formigas

Similar ao PSO, o algoritmo colônia de formigas (ACF) em vez de envolver suas características genéticas, envolve o seu comportamento social. Este algoritmo tem na cooperação, um componente chave para projetos de algoritmos. Aloca recursos computacionais para um conjunto de agentes relativamente simples (formigas artificiais) que se comunicam diretamente. Uma técnica probabilística, baseado na inteligência de enxame, muito poderosa para resolver problemas computacionais de otimização.

2.3.2.5. Hill Climbing

Este algoritmo é amplamente aplicado na inteligência artificial para alcançar um estado objetivo de nó inicial. HC, Sendo um algoritmo evolucionário partilha as vantagens como eficiência do tempo de computação inicialização, etc.

2.3.2.6. Tabu Search (TS)

Junto com SA é uma das técnicas mais importantes para problemas de otimização combinatória. Com a TS, a complexidade não está presente apenas nos problemas mas na própria técnica, consideraram os autores (Hertz, Taillard, & Werra, 1995). Isto claro, não é uma vantagem mas uma vez que a situação pode ser vista como uma etapa exploratória num campo diferente com novas técnicas que provavelmente pode se tornar mais elegante e talvez mais simples quando a memória de inteligência artificial e exploração serem mais bem integrados.

2.4. Conclusões

Relativamente aos diferentes modelos de semicondutores descritos ao longo deste texto foram sendo apresentadas abordagens que permitem sustentar a conclusão deste capítulo. Entre os modelos abordados, destaca-se o modelo híbrido de elementos finitos, modelos Lumped-charge e os modelos analíticos que tem emergido e também têm sido bem-sucedidos.

A revisão do estado de arte realizada permitiu reconhecer a importância dos métodos de modelação dos semicondutores, a sua aplicação, os algoritmos de otimização aplicado nas mais diversas áreas.

O modelo híbrido Método de Elementos Finitos comparado com outros métodos permite uma implementação fácil num curto espaço de tempo utilizando simuladores de circuitos elétricos de uso comum como PSPICE, é um método adequado para soluções computacionais como é o caso deste trabalho e com poucos elementos pode simular problemas complexos.

No campo da engenharia, a aplicação nos projetos de eletrónica de potência, os requisitos fundamentais exigidos estão relacionados com o grau de precisão versus velocidade do cálculo, que deve ser fornecido por modelos úteis para simulação de circuito, deve ser especificado (Kraus & Mattausch, 1998b). O conceito de modelação exige altos graus de precisão por isso é necessário o conhecimento profundo de dispositivos de potência individual.

Por outro lado, estimação de parâmetros dos modelos constitui também uma das questões críticas para modelos semicondutores, que doravante exige atenção. Neste trabalho, apresentamos a abordagem proposta por Araújo (2008b), que resolve equação de difusão

ambipolar utilizando a formulação variacional posteriormente seguida de uma solução aproximada por elementos finitos. Esta abordagem apresentada baseada na EDA permite implementação fácil dentro de simuladores padrões como SPICE, descrevendo a distribuição de portadores de cargas nas zonas fracamente dopadas, aquilo que é público como uma das principais preocupações na descrição precisa do comportamento dinâmicos dos semicondutores de potência. A acrescentar ainda que a abordagem da EDA apresentada pelo autor acima é muito promissora, uma vez que apresenta um bom compromisso entre o tempo de cálculo e a precisão obtida, umas das questões importantes na modelação de semicondutores de potência baseado na física.

O processo de otimização é arquitetado em ficheiros m.file, um dos componentes do MATLAB, utilizando técnicas de otimização de enxames de partículas e simulated annealing que posteriormente são analisados os desempenhos na extração de parâmetros semicondutores de potência como o diodo.

Nos capítulos seguintes estes conceitos serão aplicados na modelação de díodos de potência.

3. MODELAÇÃO DE DÍODOS DE POTÊNCIA BASEADA NO MEF

3.1. Nomenclatura

A_e – Área de elemento (cm^2)

$C_{io}, C_{jo}, C_{ij}, C_{ji}$ – Capacitância Elementar (F)

D_n, D_p – Constante de difusão de elétrons e lacunas

ε – Constante dielétrico de silício (F/cm)

I_d – Corrente de díodo

I_n, I_p – Corrente de elétron e lacuna (A)

J_n, J_p – Densidades de correntes de lacuna e elétron (A/cm^2)

L_f – Indutância de fuga de circuito (H)

L_{E_e} – Largura de elemento e

μ_n, μ_p – Mobilidade de elétron e lacuna (cm^2/Vs)

N_B – Concentração de doping na base (cm^{-3})

n_i – Concentração de portadores intrínseco (cm^{-3})

p, n – Concentração elétron / lacuna (cm^{-3})

q – Carga Elétrica (C)

R_b – Resistência Base (Ω)

$R_{io}, R_{jo}, R_{ij}, R_{ji}$ – Resistência elementar (Ω)

τ – Tempo de vida base (s)

U_l, U_r – Quebra de tensão nas junções esquerda e direita (V)

U_{res} – Quebra de tensão resistivo na base (V)

U_{sc} – Quebra de tensão de carga no espaço (V)

V_{cc} – Tensão de circuito (V)

V_T – Tensão térmica (V)

X_i, X_r – fronteira direita, esquerda na base (cm)

W – largura de base de diodo (cm)

$Wb = X_r - X_l$ – Largura instantâneas da base (cm)

3.2. Introdução

O diodo de potência é, entre todos os dispositivos de comutação estáticos utilizados na aplicação de eletrônica de potência, talvez o mais simples. Este dispositivo desempenha uma função preponderante nos circuitos de eletrônica de potência, por isso, é também considerado um dos mais importantes. Não existe um modelo único de diodo de potência porque existe variados simuladores e requisitos, apesar de vários modelos propostos.

Geralmente, modelos de potência são classificados como macro modelos ou modelos físicos detalhados. Os macros modelos são formados por circuitos elétricos que não estão diretamente relacionado com o processo físico interno no dispositivo enquanto os modelos físicos incluem equações para deriva e difusão de elétrons e lacunas. Esses modelos contêm equações matemáticas e parâmetros que são complicados incorporar dentro dos simuladores de circuitos.

Como mencionado no parágrafo acima, diodo de potência é amplamente utilizado nos campos de eletrônica de potência. Quando um diodo apresenta tensão de bloqueio grande, a necessidade da camada de depleção ser larga para poder restringir a intensidade do campo elétrico máximo abaixo do nível de impacto de ionização. Portanto, essas duas exigências serão satisfeitas num diodo de junção p-n fracamente dopado com largura suficiente para acomodar a camada de depleção exigido.

Para além das duas camadas p e n, o diodo de potência apresenta uma terceira camada, conhecida como zona n-. Esta camada extra, localizada no meio das duas camadas convencionais, com baixa dopagem, é responsável no aumento da capacidade do componente quando aplicado em altas tensões. Esta camada acrescenta a capacidade

resistiva do díodo quando está em condução. Portanto, podemos afirmar que o díodo de potencia, baseado na sua constituição, face ao seu campo de aplicação, apresenta as seguintes necessidades (Araújo, 1998):

1. O bloqueio das tensões elevadas. Apresenta um gama de tensões na ordem de centena de Volt a alguns kVolt, acrescentado a isso, a existência de uma região extensa e pouco dopada, nas mediações da camada de depleção.
2. Baixa tensão para suportar passagens correntes elétricas elevadas. Sendo a base uma zona muito resistiva, essa condição origina a injeção de eletrões livres e lacunas, para aumentar a sua capacidade de condução. Para conseguir este efeito, foram introduzido dois emissores nos dois extremos da zona citada. Os emissores são P+ e N+.
3. Baixas perdas de comutação. Esta característica está ligada ao tempo médio de vida e ao próprio circuito em que está inserido.

Resumidamente, o díodo de potência é constituído pelas camadas p+ n- n+, como apresentado na figura. Assim, nos pontos que se segue, serão apresentados os principais modelos para várias regiões que constituem o díodo de potência.

- Modelos para camadas de base (n⁻)
- Modelos para as camadas emisoras fortemente dopadas p⁺/n⁺.
- Modelos para as quedas de tensão:

3.3. Modelo para camadas de Base (n⁻)

3.3.1. Modelo para a equação de difusão ambipolar

Em (Araújo, 2007, 2008b), demonstrou-se ser possível resolver a EDA, utilizando uma formulação variacional, para poder obter solução baseada no método de elemento finito. Para recapitular, nos parágrafos em baixo vamos faz-se um breve resumo sobre o método. Como vimos a fórmula geral alcançada no capítulo 2, o seguinte sistema de equações:

$$[M] \left[\frac{\partial p}{\partial t} \right] + [K][p] + [F] = [0] \quad (3.1)$$

onde,

$$M = \int_{x_l}^{x_r} \frac{1}{D} N^T N dx$$

$$K = - \int_{x_l}^{x_r} \frac{\partial N^T}{\partial x} \frac{\partial N}{\partial x} dx + \int_{x_l}^{x_r} \frac{N^T N}{D\tau} dx \quad (3.2)$$

$$F = -[(f + g)N^T]_{x_l}^{x_r}$$

com,

$$[M] = \sum_{e=1}^n [M_e]$$

$$[G] = \sum_{e=1}^n G_e \quad (3.3)$$

$$[F] = \sum_{e=1}^n [F_e]$$

Associando cada elemento finito as matrizes elementares, teremos:

$$M_e = \frac{A_e l_e}{6D} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad (3.4)$$

$$G_e = \frac{A_e}{l_e} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \frac{A_e l_e}{6D\tau} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \quad (3.5)$$

Com L_{Ei} definida apenas no primeiro e últimos elementos:

$$L_{Ei} = [-(f - g)A_1 \ 0]^T \quad (3.6)$$

$$L_{En} = [0 \ -(f + g)A_n]^T$$

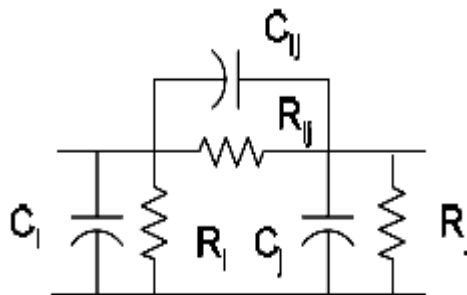


Figura 3. 1 - Circuito elétrico equivalente a um elemento finito

A equação (3.1) é resolvida fazendo analogia com um sistema de equações de uma rede RC como se segue:

$$C_{ij} = -\frac{A_e l_e}{6D}; C_i = C_j = \frac{A_e l_e}{2D} \quad (3.7)$$

$$R_{ij} = \frac{6D\tau l_e}{6D\tau A_e - A_e l_e^2}; R_i = R_j = \frac{2D\tau}{A_e l_e^2}$$

Cada uma das malhas RC corresponde a um elemento finito. Associando essas malhas RC em série, juntamente com duas fontes de corrente para a implementação das condições fronteira, o resultado será um sistema de equações:

$$C \left[\frac{\partial V}{\partial t} \right] + G[V] + I = [0] \quad (3.8)$$

análogo à equação (3.1), em que C, G e I correspondem a M, K e F respetivamente.

Desta forma, a solução EDA na região n- pode ser obtida por meio da solução de circuito elétrico da Figura 3.2.

Posteriormente, serão construídos os subcircuitos para a implementação dessas relações.

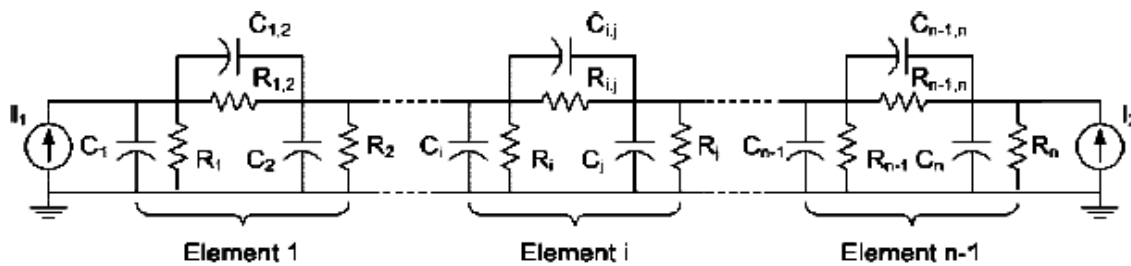


Figura 3. 2 - Circuito elétrico equivalente da EDA pelo MEF

3.4. Modelação das zonas p⁺ e n⁺

Depois de estabelecer o subcircuito elementar relacionado a cada elemento finito da zona n-, correspondente à parte associada às matrizes elementares [Me] e [Ke], a solução do sistema de equações 3.1, equivalente à solução da equação de difusão ambipolar, exige a inclusão das condições fronteiras associadas à matriz [F]. Estas condições são equivalentes a duas fontes de corrente, injetadas no primeiro e últimos nós:

$$I_l = -(f + g)A_l = \frac{\partial p}{\partial x|_{x=x_l}} A_l = \left[-\frac{J_{n|x=x_l}}{2qD_n} + \frac{J_{p|x=x_l}}{2qD_p} \right] \times A_l \quad (3.11)$$

$$I_r = -(f + g)A_r = \frac{\partial p}{\partial x|_{x=x_r}} A_r = \left[-\frac{J_{n|x=x_r}}{2qD_n} + \frac{J_{p|x=x_r}}{2qD_p} \right] \times A_r \quad (3.12)$$

onde A_l e A_r são os valores das áreas associadas ao primeiro e ao último elementos finitos e X_l e X_r são os limites da zona n^- . A necessidade de adotar um modelo para as zonas emissoras p^+ e n^+ quando se pretende calcular as condições fronteiras. As correntes de recombinação $J_{n|x=x_l}$ e $J_{p|x=x_r}$ dependem das concentrações dos bordos, $p_{|x=x_l}$, $n_{|x=x_l}$, $p_{|x=x_r}$ e $n_{|x=x_r}$. Como $p \approx n$ vem:

$$J_{n|x=x_l} = qh_p p_{|x=x_l}^2 \quad (3.13)$$

$$J_{p|x=x_r} = qh_n p_{|x=x_r}^2 \quad (3.14)$$

De uma forma geral:

$$\frac{\partial p}{\partial x} = \frac{I_n - \frac{\mu_n}{\mu_p} I_p}{2AK_T T \mu_n} = \frac{I_n}{2qAD_n} - \frac{I_p}{2qAD_p} \quad (3.15)$$

$$\frac{\partial p}{\partial x} = \frac{1}{2qA} \left[\frac{I_n D_p - I_p D_n}{D_n D_p} \right] \quad (3.16)$$

$$I_T = I_n + I_p \quad (3.17)$$

com

$$I_{p|x=x_l} = I_T - qA_l h_p p_{|x=x_l}^2; I_{n|x=x_l} = qA_l h_p p_{|x=x_l}^2 \quad (3.18)$$

$$I_{n|x=x_r} = I_T - qA_r h_n p_{|x=x_r}^2; I_{p|x=x_r} = qA_r h_n p_{|x=x_r}^2 \quad (3.19)$$

As correntes finais serão:

$$I_l = -(f + g)A_l = \frac{\partial p}{\partial x|_{x=x_l}} A_l = -\frac{1}{2q} \left[\frac{qA_l h_p p_{|x=x_l}^2 - [I_T - qA_l h_p p_{|x=x_l}^2] D_n}{D_n D_p} \right] \quad (3.20)$$

$$I_r = -(f + g)A_r = \frac{\partial p}{\partial x|_{x=x_r}} A_r = -\frac{1}{2q} \left[\frac{[I_T - qA_r h_n p_{|x=x_r}^2] D_p - qA_r h_n p_{|x=x_r}^2 D_n}{D_n D_p} \right] \quad (3.21)$$

Com o conhecimento das concentrações nos bordos e da corrente total aos terminais dos díodos conseguem determinar esses valores

3.5. Modelo para as quedas de tensão

No díodo, as quedas de tensão são calculadas através da soma das seguintes parcelas:

- As quedas de V_{JR} e V_{JL} nas junções p+n- e n- n+;
- A queda V_{RES} ao longo da zona n- modulada em condutividade;
- A queda de tensão V_{SC} suportada pela zona de depleção durante a fase de recuperação inversa.

3.5.1. Modelo relacionado ao cálculo das quedas de tensão V_{JR} e V_{JL}

Nas junções, as tensões são calculadas baseados nas equações de Boltzmann.

Na junção p⁺n⁻ vem:

$$V_{JL} = V_T \ln \left(\frac{p_{n-wp}}{p_{n_0^-}} \right); p_{n_0^-} \times n_{n_0^-} = n_i^2; n_{n_0^-} = N_b \quad (3.22)$$

$$V_{JL} = V_T \ln \left(\frac{p_{n-(wp) \times N_b}}{n_i^2} \right) \quad (3.23)$$

Na junção n⁻n⁺ vem:

$$V_{JR} = V_T \ln \left(\frac{n_{n-(W_b-W_n)}}{n_{n_0^-}} \right); p_{n_0^-} \times n_{n_0^-} = n_i^2; n_{n_0^-} = N_b \quad (3.24)$$

$$V_{JR} = V_T \ln \left(\frac{n_{n-(W_b-W_n)}}{N_b} \right) \quad (3.25)$$

Onde $p_{n_0^-}$, $n_{n_0^-}$ são os valores de concentração de lacunas e elétrons livres nas zona n- e $p_{n-(W_p)}$, $n_{n-(W_b - W_n)}$ são valores das concentrações de lacunas e electrões livres nos extremos direito e esquerdo da mesma zona e no modelo são retratados pelas concentrações nos bordos.

3.5.2. Modelo relacionado com cálculo de V_{RES}

Na zona resistiva n-, a queda de tensão, V_{RES} , relativamente a condutividade vem na forma:

$$V_{RES} = \int_{x_l}^{x_r} \frac{J_n}{q\mu_n n} dx \quad (3.26)$$

Onde $J = J_n + J_p$ e substitua-se J_n na equação:

$$p \frac{J_n}{D_p} - n \frac{J_p}{D_p} = q \frac{\partial(pn)}{\partial x} \quad (3.27)$$

vem,

$$\int_{X_l}^{X_r} \left[\frac{\frac{nJ}{D_p} + q \frac{\partial(pn)}{\partial x}}{q \mu_n n \left(\frac{p}{D_n} + \frac{n}{D_p} \right)} \right] dx \quad (3.28)$$

que se divide em duas parcelas:

$$V_{DB} = \int_{X_l}^{X_r} \left[\frac{q \frac{\partial(pn)}{\partial x}}{q \mu_n n \left(\frac{p}{D_n} + \frac{n}{D_p} \right)} \right] dx = \int_{X_l}^{X_r} \left[\frac{\frac{\partial(pn)}{\partial x}}{n \left(\frac{\mu_n p}{D_n} + \frac{\mu_n n}{D_p} \right)} \right] dx \quad (3.29)$$

$$V_{\Omega} = \int_{X_l}^{X_r} \left[\frac{\frac{nJ}{D_p}}{q \mu_n n \left(\frac{p}{D_n} + \frac{n}{D_p} \right)} \right] dx = \int_{X_l}^{X_r} \left[\frac{J}{q(\mu_n p + \mu_n n)} \right] dx$$

V_{BD} , conhecida como tensão de Deber (Deber, 1996), pode ser calculada como:

$$V_{DB} = \int_{X_l}^{X_r} \left[\frac{q \frac{\partial(pn)}{\partial x}}{q \mu_n n \left(\frac{p}{D_n} + \frac{n}{D_p} \right)} \right] dx \approx \int_{X_l}^{X_r} \left[\frac{\frac{\partial(p^2)}{\partial x}}{\mu_n \left(\frac{D_p + D_n}{D_n D_p} \right) p^2} \right] dx = \frac{\mu_n D}{2} \ln[p_{X_r} - P_{X_l}] \quad (3.30)$$

Podemos constatar que V_{BD} é dependente das diferenças das tensões nos pontos extremos da zona resistiva n^- .

A tensão V_{Ω} pode ser percebida como uma queda de tensão puramente resistiva, $V_{\Omega} = R_d \times I$, relacionada com resistividade da zona n^- , R_d . Na zona n^- , o perfil de dopagem é assumido como uniforme, portanto a formula final para o cálculo R_d é dado por:

$$R_d = \sum_{e=1}^r \left[\frac{2L_{Ee}}{q A_e [(P_{e+1} + P_e)(\mu_p \mu_n) + N_b \mu_n]} \right] \quad (3.31)$$

3.5.3. Modelo relacionado com a queda de tensão na zona de depleção

Este modelo é fundamentado na equação de Poisson:

$$\frac{\partial^2 V}{\partial^2 x} = -\frac{\partial E}{\partial x} = \frac{q}{\varepsilon} (p - n + N_b(x)) \quad (3.32)$$

Consoante o conhecimento do perfil de dopagem, assim será a integração da zona n⁻, $N_b(x)$. As situações mais acessíveis são o perfil abrupto, em que $N_b(x) = N_b = \text{constante}$ e o perfil duplamente abrupto no qual $N_b(x) = N_{b1}$ e $X_l < x < W_{b1}$ e $N_b(x) = N_{b2}$ e $W_{b1} < x < X_r$.

Portanto, se integramos a equação 3.32 teremos:

$$V_{SC} = \frac{qN_b}{2\varepsilon} X_{SC}^2 + \frac{|I_T|}{2\varepsilon A v_l} X_{SC}^2 \quad (3.33)$$

sujeito à variação da largura da zona de depleção X_{SC}^2 . Quando não é conhecido o perfil de dopagem, $N_b(x)$, será utilizado a equação 3.33.

Para a situação em que temos um perfil duplamente abrupto temos:

$$V_{SC} = \frac{qN_b}{2\varepsilon} X_{SC}^2 + \frac{|I_T|}{2\varepsilon A v_l} X_{SC}^2, X_l < X_{SC} < X_{b1} \quad (3.34)$$

$$V_{SC} = \frac{qN_{b1}}{2\varepsilon} X_{SC}^2 + \frac{qN_{b2}}{2\varepsilon} [X_{SC} - W_{b1}]^2 + 2[X_{SC} - W_{b1}] + \frac{|I_T| X_{SC}^2}{2\varepsilon A v_l}, W_{b1} < X_{SC} < X_r \quad (3.35)$$

3.5.4. Cálculo das fronteiras na zona de depleção

Para obter o sistema de equações de difusão ambipolar será necessário calcular a largura de cada elemento finito. Esta largura é constante e depende apenas da largura total da zona n⁻ e W_b , na fase de condução direta.

Quando a concentração de portadores, $p(x)$, situado nos extremos da zona n⁻, que neste trabalho são denominados por X_l e X_r , tendem a tornar negativa, anula-se as restrições do problema, relacionado com as fontes de corrente nos extremos dos nós, I_l e I_r , persistem na averiguação da condição para serem aplicadas nos X_l e X_r e com condição acrescentada de que a concentração $p(X_l)$ e $p(X_r)$ seja segura como nula (Araújo, 2008b).

Assim é necessário um esboço de cálculo que garanta todas essas condições supracitadas. Uma análise precisa da solução, partindo do momento em que as concentrações tendem para valores inferior a zero, resume-se a deslocar a fronteira em causa, anular a concentração e ajustar os sucessivos valores da abcissa até atingir a convergência.

Concentrações relatadas tendem a ser negativas, por isso, o seu valor é multiplicado por um ganho elevado e posteriormente deduzido à totalidade do valor da largura da zona n-.

3.6. Implementação no código fonte IsSpice

Para a realização desta tese foi utilizada a ferramenta IsSpice4, quarta geração da Intusoft, desenvolvido pela Berkeley SPICE 3F.5, mecanismo de simulação combinado com o simulador XSPICE da Georgia Tech.

3.6.1. Circuito relacionado as resistências RJ, RI e RIJ

Na figura 3.3, está ilustrado o subcircuito responsável pela simulação das resistências variáveis Ri e Rj do deporto elementar. As relações deste subcircuito estão implementadas entre os nós 1 e 3:

$$R_{eq} = \frac{V}{I}; V(I) = \frac{2D\tau}{A_e} \times \frac{I}{I(V_{cnt})} \Rightarrow R_{eq} = \frac{2D\tau}{A_e I(V_{cnt})} \quad (3.9)$$

$$L_{E_e} = I(V_{cnt}) \Rightarrow R_{eq} = \frac{2D\tau}{A_e L_{E_e}} \quad (3.10)$$

utilizando a fonte de tensão B1 (controlada entre nós 2 e 3) que controlada pela corrente que a atravessa (medida pela fonte de tensão V1, de valor zero, entre os nós 1 e 2). Ao mesmo tempo os valores Ae, D e τ passam-se como parâmetros para circuito. A largura de elemento finito, L_{E_e} , representa-se pela corrente, $I(V_{cnt})$, injetada no nó 4 e medida na fonte VCNT.

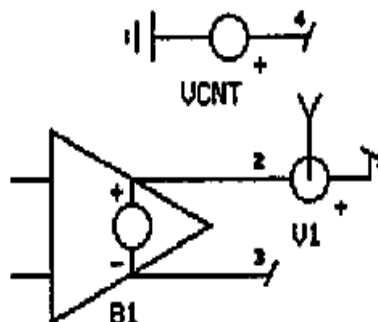


Figura 3. 3 - Subcircuito responsável pela simulação de uma resistência variável

No IsSpice o subcircuito é descrito como se segue:

```
*****  
  
*SYM=R_VAR_1 1 3 4  
  
* DWG=C:\ASA\SPICE\CIR\R_VAR_1.DWG  
  
.SUBCKT R_VAR_1  
  
*ESTE SUBCIRCUITO SIMULA UMA RESISTÊNCIA VARIÁVEL  
  
* 1 E 3 SÃO OS NÓS DA RESISTÊNCIA  
  
* 4 É O NÓ DE CONTROLO DA LARGURA DO ELEMENTO FI  
  
B1 2 3 V= {2*D*TAU} *I(V1) / ({AE}*I(VCNT))  
  
V1 1 2 DC 0  
  
.ENDS  
  
*****
```

Os códigos acima, são referidos as resistências variáveis Ri e Rj.

Relativamente a resistência variável Rij, o circuito projetado é idêntico com exceção a descrição do código que é o seguinte:

```
*****  
  
* O CIRCUITO SIMULA UMA RESISTÊNCIA VARIÁVEL  
  
* SYM= R_VAR_2 1 3 4  
  
* .SUBCKT R_VAR_2  
  
* 1 3 SÃO NÓS DA RESISTÊNCIA  
  
* 4 É O NÓ DE CONTROLO DA LARGURA DO ELEMENTO FINITO LEE  
  
B1            2            3            V={6*D*TAU*AE} *I(VCNT) *I(V1) / (6*{D*TAU*AE})-  
{AE} *I(VCNT) *I(VCNT) *I(VCNT)
```

V1 1 2 DC 0

VCNT 4 DC 0

.ENDS

3.6.2. Circuito relacionado aos condensadores C_i, C_j e C_{ij} .

Para as capacidades variáveis C_i e C_j , conforme está projetado na Figura 3.4, os subcircuitos a desenvolvido nos nós 4 e 1, estão relacionados:

$$C_{eq} = \frac{I_{41}}{\frac{\partial V_{41}}{\partial t}}; I_{41} = I_{R_i} + I_{C_i} + I_{B_1} \approx I_{C_i} + I_{B_1} \text{ se } I_{R_i} \ll I_{C_i}, I_{B_1} \quad (3.11)$$

$$I_{C_i} = I(V_1) = C_i \frac{\partial V_{41}}{\partial t} = C_i \frac{\partial V_{31}}{\partial t} \quad (3.12)$$

$$I_{B_1} = I(V_1)[I(V_{cnt} - 1)] \Rightarrow I_{41} = I(V_{cnt}) \times C_i \frac{\partial V_{31}}{\partial t}$$

$$C_i = \frac{A_e}{2D}; I(V_{cnt}) = L_{E_e} \Rightarrow I_{41} = \frac{A_e L_{E_e}}{2D} \frac{\partial V_{31}}{\partial t} \Rightarrow C_{eq} = \frac{A_e L_{E_e}}{2D} \quad (3.13)$$

em que as duas correntes controla a fonte de corrente B_1 (localizada entre os nós 4 e 1):

- A corrente associada à capacidade de referência C_i (controlada entre nós 3 e 1), determinada pela fonte V_1 (colocada entre nós 4 e 3)
- A corrente controlada pela fonte de controlo, V_{CNT} , caracterizada pela largura de elemento finito, L_{E_e}

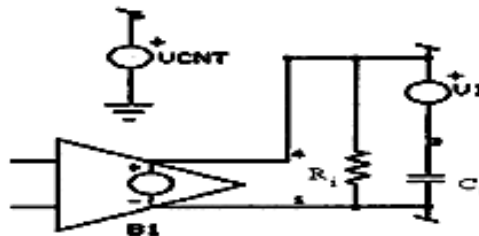


Figura 3. 4 - Subcircuito da capacidade variável

Os restantes valores dos parâmetros A_e , D , e τ são passados como parâmetros para o circuito.

* O CIRCUITO SIMULA UMA RESISTÊNCIA VÁRIAVÉL

* DWG=C:\SPICE\CIR\C_VAR_1.DWG

.SUBCKT C_VAR_1

* ESTE CIRCUITO SIMULA A CAPACIDADE VARIÁVEL*

* 1 E 4 SÃO OS NÓS DA CAPACIDADE

* 5 É O NÓ DE CONTROLO DA LARGURA DO ELEMENTO FINITO LEE

VCNT 5 0 DC 0

CI 3 1 { $AE/(2*D)$ }

V1 4 3 DC 0

RI 4 1 I=[I(VCNT)-1]*I(V1)

.ENDS

Capacidade variável C_{ij} também é semelhante com a exceção ao código utilizado que é descrito como:

*SYM=C_VAR_2 1 4 5

*DWG=C:\ASA\SPICE\CIR\C_VAR_2.DWG

.SUBCKT C_VAR_2

* ESTE SUBCIRCUITO SIMULA UMA CAPACIDADE VARIÁVEL

*1 E 4 SÃO OS NÓS DA CAPACIDADE

*5 É O NÓ DE CONTROLO DA LARGURA DO ELEMENTO FINITO LEE

VCNT 5 0 DC 0

CI 3 1 {-AE/(6*D)}

V1= 4 3 DC 0

RI 4 1 1 MEG

B1 4 1 I=[I(VCNT)-1]*I(V1)

.ENDS

3.6.3. Circuito relacionado aos condensadores II e Ir

As correntes I_l e I_r , foram divididas em duas componentes para a construção do circuito:

$$I_l = -\frac{1}{2q} \left[\frac{qA_l h_p p_{|x=X_l}^2}{D_n} + \frac{qA_l h_p p_{|x=X_l}^2}{D_p} - \frac{J_T}{D_p} \right] = \frac{I_T}{2qD_p} - \frac{A_l h_p p_{|x=X_l}^2}{D} \quad (3.14)$$

$$I_r = -\frac{1}{2q} \left[\frac{I_T}{D_n} - \frac{qA_r h_n p_{|x=X_r}^2}{D_n} - \frac{qA_r h_n p_{|x=X_r}^2}{D_p} \right] = -\frac{I_T}{2qD_n} + \frac{A_r h_n p_{|x=X_r}^2}{D} \quad (3.15)$$

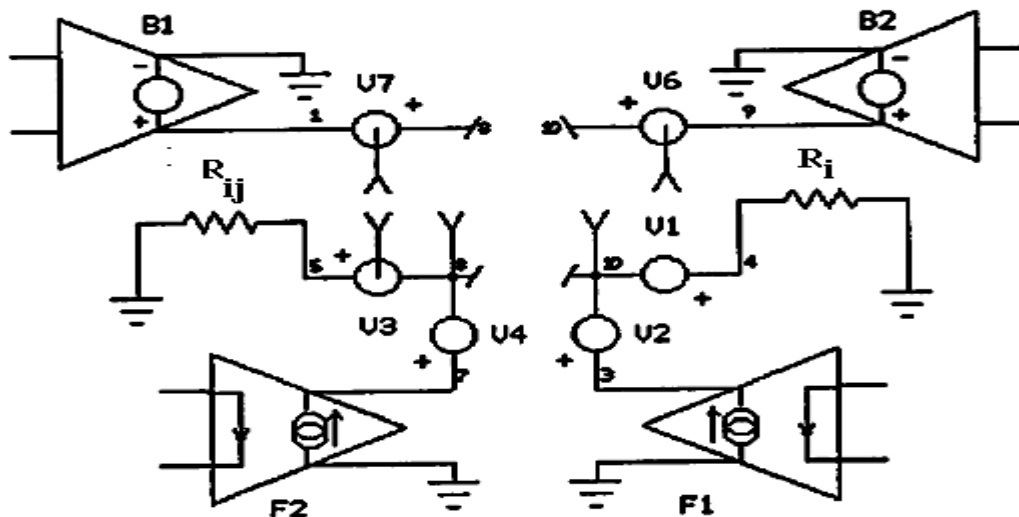


Figura 3. 5 - Esquema do circuito utilizado para simulação das zonas emissoras.

Na Figura 3.5 está ilustrada a forma como foi implementada o esquema no simulador. As tensões no primeiro e último nó, controlam as fontes de corrente B1 e B2 e desenvolvem a relação do segundo termo das equações (3.14) e (3.15). A totalidade das correntes que atravessam o díodo são responsáveis pelas fontes de corrente F1 e F2, medida através da tensão V5, que não apareça na Figura 3.5, implementam os primeiros termos das supramencionadas equações. Já as medidas das resistências R_i e R_{ij} são realizadas por via da corrente relacionada ao cálculo da largura dos elementos finitos ao longo da recuperação inversa.

O código associado ao circuito é o seguinte:

*V5 MEDE A CORRENTE TOTAL

*V8 É A CONCENTRAÇÃO NO PRIMEIRO NÓ

*(V10) É A CONCENTRAÇÃO NO ÚLTIMO NÓ

V1 4 10 DC 0

V2 3 10 DC 0

V3 5 8 DC 0

V4 7 8 DC 0

V7 8 1 DC 0

V6 10 9 DC 0

R1 4 0 0.1MEG

R2 0 5 0.1MEG

F1 0 3 V5 1.25

F2 0 7 V5 3.125

B1 1 0 I=2.1*V(8)*V(8)

B2 9 0 I=2.1*V(10)*V(10)

3.6.4. Circuito relacionado com as quedas de tensão VJL e VJR

As quedas de tensão são implementadas recorrendo as duas fontes de tensão, controladas pelas tensões nos primeiros e últimos nós, tal como aparece na Figura 3.6, através do seguinte código:

```
*VJR=VT*LOG(PXR/ND)
```

```
B4 36 0 V=V(10)>0? 26E-3*LN((V(10)*1E17+1)/1E14): 0
```

```
*VJL=VT*LOG(PXL*ND/NI^2)
```

```
B3 35 0 V=V(8)>0? 26E-3*LN((V(8)*1E17*1E14+1)/1.44E20): 0
```

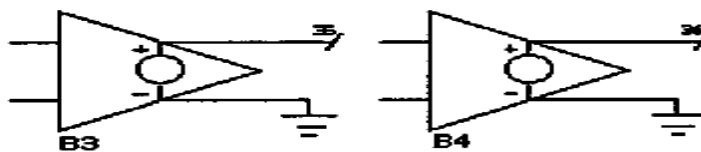


Figura 3. 6 - Circuito responsável pela simulação das quedas de tensão V_{JL} e V_{JR}

3.6.5. Circuito relacionado com as quedas de tensão VRES

No circuito da Figura 3.7, a tensão associada é resultante da soma das fontes de tensão dependentes das tensões nos nós e da largura dos elementos finitos.

O código para esta implementação é o seguinte:

```
ALIAS V(47)= RTOTAL
```

```
.PRINT DC V(47)
```

.PRINT TRAN V(47)

B3 43 0 V=V(8)>0? I(V21)/((V(8)+V(25))*1E3*1347+1934): I(V21)/1934

* LE1/(2*PAV1*NIUNP+NIUN*ND)

* B10 44 0 V=V(25)>0? I(V22)/((V(25)+V(37))*1E3*1347+1934): + I(V22)/1934

* LE2/(2*PAV2*NIUNP+2*NIUNP*ND)

B4 45 0 V V=V(37)>0? I(V23)/((V(37)+V(38)+V(39))*1E3*1347+1934): I(V23)/1934

*LE3/(2*PAV3*NIUNP+2*NIUNP*ND)

B5 46 0 V=V(38)>0? I(V24) /((V(38)+V(39))*1E3*1347+1934) : I(V24)/1934

*LE4/(2*PAV4*NIUNP+2*NIUN*ND)

B7 48 0 V=V(39)>0? I(V25)/((V(39)+V(13))*1E3+1347+1934) : I(V25)/1934

* LE5/(2*PAV5*NIUNP+2*NIUNP*ND)

B8 49 0 V=V(13)>0? I(V32)/(((13)+V(51))*1E3*1347+1934) : +I(V32)/1934

* LE6/(2*PAV6*UNIUNP+2*NIUNP*ND)

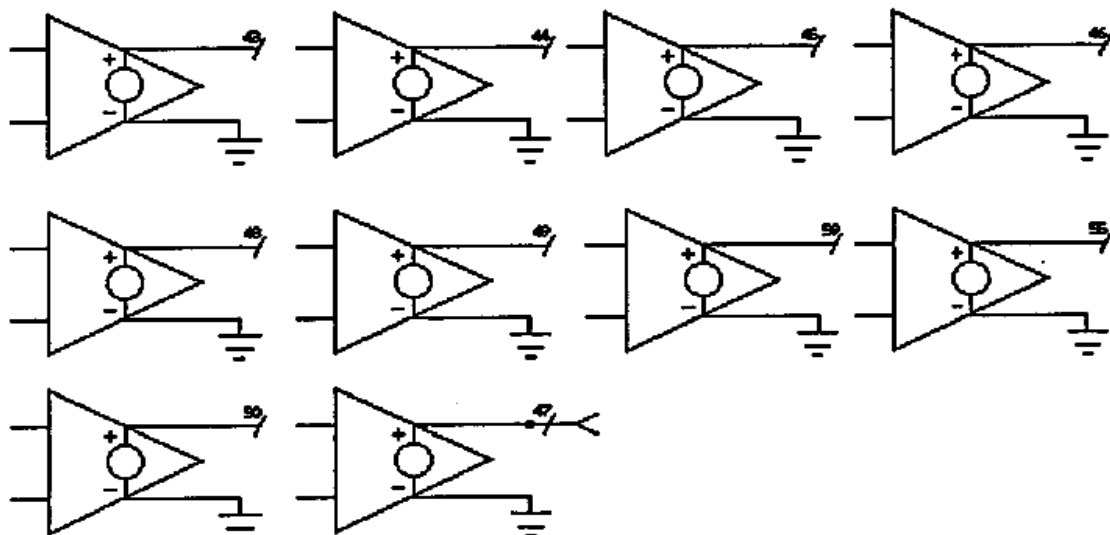


Figura 3. 7 - Esquemas de circuitos associados à queda de tensão VRES

```
B11 59 0 V=V(51)>0? I(V31)/((V(51)+V(29))*1E3*1347+1934 :) + I(32)/1934
```

```
*LE7/(2*PAV7*NIUNP+2*NIUNP*ND)
```

```
.PRINT DC V(60)
```

```
B12 55 0 V=V(29)>0? I(V26)/((V(29)+v(40))*1E3*1347+1934): + I(V26)/1934
```

```
.PRINT TRAN V(60)
```

```
*LE8/(2*PAV8*NIUNP+2*NIUN*ND)
```

```
B1 60 0 V=I(V3)>0? 7.66E-8*(1E14)*(1.5E4*I(V3))^2 : - V(35) -V(36)-V(47)
```

```
B9 50 0 V=V(40)>0? I(V27)/((V(40)+V(10))*1E3*1347+1934) : + I(V27)/1934
```

```
* VSC=Q*(ND+PTR)/(2*EPSILON)*XL^2 OU VD=VJL+VJR+VRES
```

```
*LE9/(2*PAV9*NIUNP+2*NIUN*ND)
```

```
*****
```

```
B6 47 0 V V=2/(1.6E-5*0.04)*(V(43)+V(44)+V(45)+V(46)+V(48))+V(49)+V(59)+
```

```
V(55)+V(50)
```

```
* RD TOTAL
```

3.6.6. Circuito relacionado com as quedas de tensão VSC

O modelo implementado no IsSpice para simular este circuito em que a largura da zona de depleção, XSC, controla a fonte de tensão, tal como está representada na Figura 3.8.

O código para este circuito é o seguinte:

```
*****
```

```
* ALIAS V(60)
```

```
.PRINT DC V(60)
```

```
.PRINT TRAN V(60)
```

```
B1 60 0 V=I(V3)>0? 7.66E-8*(1E14)*(1.5E4*I(V3))^2 : - V(35) -V(36)-V(47)
```

```
* VSC=Q*(ND+PTR)/(2*EPSILON)*XL^2 OU VD=VJL+VJR+VRES
```

```
*****
```

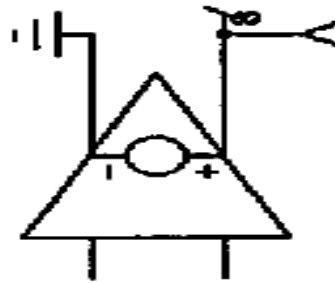


Figura 3. 8 - Esquema do circuito da queda de tensão na zona de depleção

3.6.7. Circuito associado ao cálculo de XSC

O cálculo deste circuito foi baseado na Figura 3.9, em que a fonte de corrente B1 foi responsável pelos cálculos das fronteiras. Para isso, foi necessário introduzir duas resistências, de elevado valor, nos extremos dos dipostos relacionados com a modelação da zona n. B1 é responsável pela medição da corrente.

Os valores das resistências de cada nó são calculados de forma a respeitar a divisão elementar do domínio. As resistências têm os valores semelhantes caso os elementos finitos sejam equidistantes

A largura dos elementos finitos L_{E_e} é alcançada pela divisão da corrente da fonte B1 por todos os ramos que compõe os elementos finitos existentes.

O código associado é:

* ALIAS (V1)=WB-WSC

* ESTA PARTE IMPLEMENTA O DIVISOR DE CORRENTES

R1 1 2 1K

R2 3 4 1K

R3 6 7 1K

R4 8 9 1K

R5 11 12 1K

R6 13 14 1K

R7 15 16 1K

*ESTA PARTE CALCULA A LARGURA DA ZONA N-, WB-WSC

B1 0 5 I=I(V10)>0? 9M-1.5E4*I(V10) : 9M

V1 5 10 DC 0

* ESTAR PARTE MEDE O TAMANHO DE CADA ELEMENTO FINITO

V2 10 1 DC 0
V3 10 3 DC 0
V4 10 6 DC 0
V5 10 8 DC 0
V6 10 14 DC 0
V7 10 12 DC 0
V7 10 13 DC 0
V8 10 16 DC 0

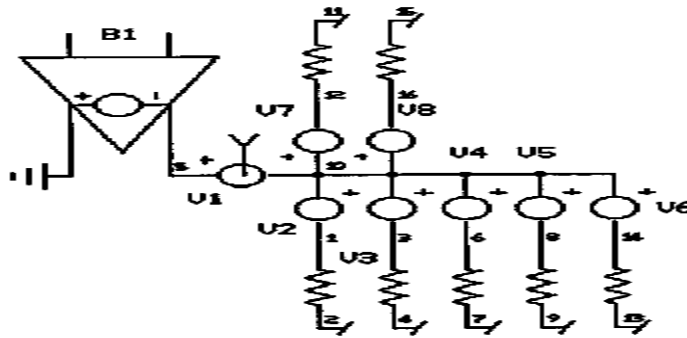


Figura 3. 9 - Circuito associado ao cálculo das fronteiras na camada de depleção

3.6.8. Circuito final obtido

O circuito final do díodo de potência obtido está ilustrado na Figura 3.10, incluindo os vários subcircuitos que foram descritos nos parágrafos acima. Conforme pode-se ver na Figura, temos:

- No bloco A, o subcircuito ligado a solução baseado na equação da EDA, que caracteriza a zona de armazenamento n^- . São nove subcircuitos que desenvolvem os dipostos elementares relacionados com as matrizes [C] e [G]. É injetada corrente relativo a à largura de cada elemento finito.
- Nos Bloco B estão implementados modelos correspondentes as zonas p^+ e n^+ .
- No bloco C é constituída por fontes controladas B3 a B11 e B16 a B18, que correspondem aos cálculos das quedas de tensão V_{RES} , V_{JL} e V_{JR} .
- O bloco D assume os encargos dos cálculos das larguras da zonas de depleção e conseqüentemente o valor da largura elementar. A responsabilidade do cálculo da largura total é atribuída a B1.
- Bloco E, formado pelas fontes B22, B23 e B21, é responsável pelos cálculos da queda de tensão relacionada com a zona de depleção.

- Por fim, o bloco F, faz as ligações com o com os elementos do circuito externo. É interpretada pela fonte de tensão controlada B24, que é a soma de todas as quedas de tensão.

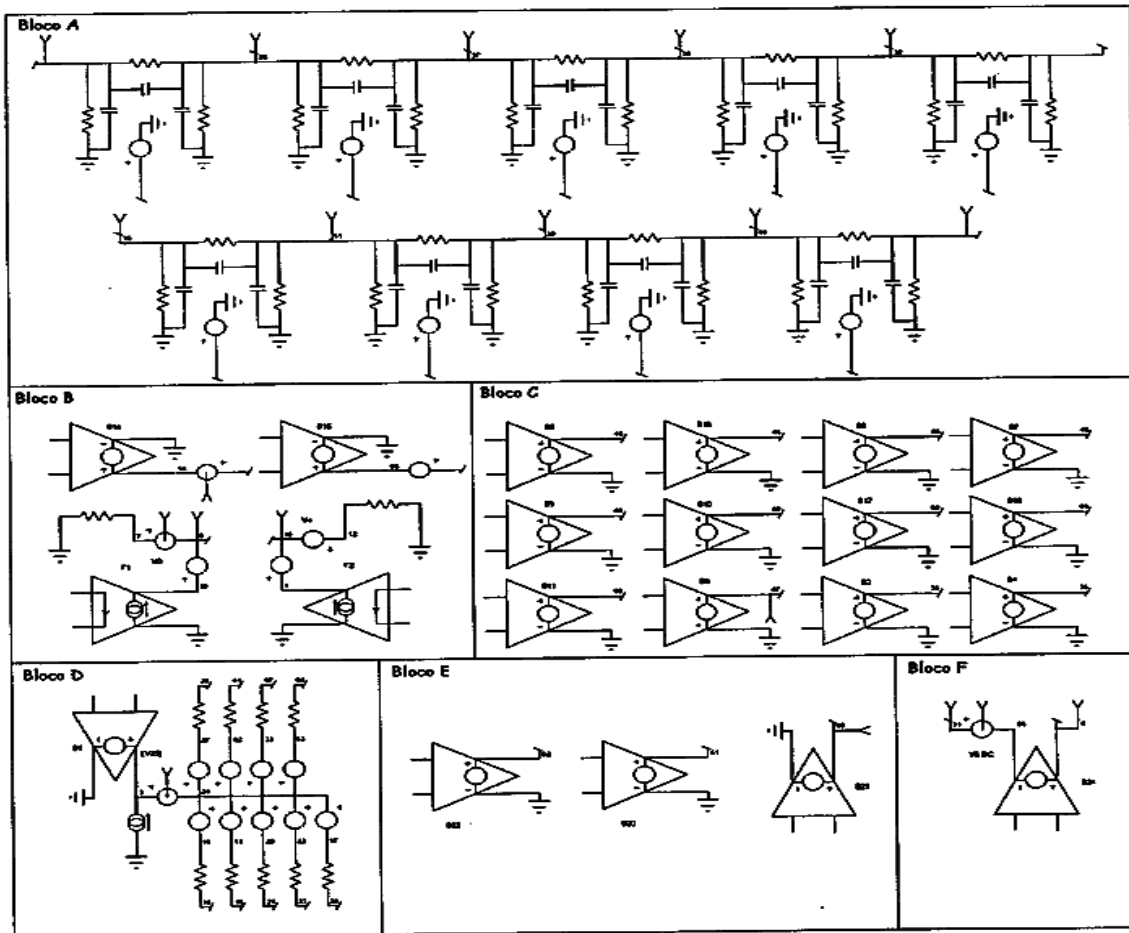


Figura 3. 10 - Modelo final do díodo de potência (Araújo, 98)

3.7. Conclusões

Ao longo do capítulo foram expostos os modelos matemáticos que caracterizam os díodos de potência. Em cada secção foi relatada as características de cada subcircuito que compõe o circuito completo do díodo, ilustrado na Figura 3.10. A maioria das equações foi implementada por meio das funcionalidades ABM do IsSPICE4⁵.

⁵ Consulta realizada no dia 20.10.2012 - <http://www.intusoft.com/lit/isspice4.pdf>

O resultado é uma solução analógica original para equação da malha RC envolvendo parâmetros variáveis. Desta forma, levando em consideração a natureza distribuída de cargas dinâmicas na região base do dispositivo, houve uma grande melhoria na precisão e capacidade do modelo. Os modelos podem ser manipulados por algum software de simulação genérica da família SPICE, com parâmetros variáveis e elementos não lineares.

A região de base (n^-), considerada como o núcleo principal é executada através de um esquema de realimentação, para permitir determinar os resultados precisos de tempo de recuperação reversa, tensão e corrente.

O circuito permite ainda através de modificação de quantidade de elementos finitos, que estão relacionados com a equação de difusão ambipolar e por acerto de números de subcircuitos, variar o compromisso entre o tempo de cálculo e a precisão. Essa mesma variação pode também ser estimulada ao acrescentar/diminuir números iguais de resistências no circuito do cálculo da largura elementar, bem como acrescentar/diminuir números iguais de subcircuitos relativos no cálculo de quedas de tensão resistiva.

Entre outras vantagens apresentadas por esta abordagem, podemos citar o facto de que a implementação do modelo baseado na física nos simuladores padrões como SPICE é simples. Dado que cada malha RC descreve uma parte da zona fracamente dopada, as suas propriedades podem ser definidas independentemente, o que é extraordinariamente útil para dispositivos com uma característica de dopagem variável ou tempo de vida variável.

Nos elementos, as propriedades dos dispositivos, D e τ , identicamente são parâmetros de subcircuitos que facilitam soluções para materiais híbridos.

O método de elementos finitos exibido neste capítulo resolve equações diferenciais descrevendo semicondutores parcialmente analítica e parcialmente numéricas expondo as suas vantagens quando comparado com os outros métodos como a alta precisão de comportamento de portadores de cargas resultante sem aumentar tempo de execução, como pode ser observado utilizando simuladores de dispositivos.

4. EXTRAÇÃO DE PARÂMETROS DO MODELO DO DÍODO DE POTÊNCIA

Na Tabela 4.1 estão listados os parâmetros pretendidos, para um modelo de diodo de potência baseado na física dos semicondutores.

Os valores dos parâmetros podem ser obtidos de diversas fontes, entre elas o procedimento empírico. Contudo, a principal aproximação continua a ser a extração através das folhas características dos dispositivos.

Hoje em dia, no desenvolvimento de projetos de eletrônica de potência, os projetistas envolvidos sentem a necessidade urgente de auxílio de um software capaz de extrair parâmetros dos modelos dos semicondutores de potência de uma forma mais precisa e eficiente. Neste trabalho propõe-se o uso de técnicas de otimização para se proceder à extração dos valores dos parâmetros.

Tabela 4. 1 - Parâmetros de modelo de díodo (Kraus & Mattausch, 1998)

Símbolo	Unidade	Parâmetro
A	cm^2	Área ativa
W_B	cm	Largura de região deriva
N_B	cm^{-3}	Concentração de dopagem da região deriva
τ_{HL}	s	Tempo de vida de nível alto
h_p	-	Parâmetro de recombinação de cátodo

A informação necessária é um conjunto de características dinâmicas e estáticas do dispositivo, que define as condições de funcionamento. A realização de otimização é feita pela comparação de resultados experimentais e simulados no IsSpice, em que o resultado da função objetivo é o valor do erro, num espaço multidimensional, em que a dimensão do espaço de pesquisa é igual ao número de parâmetros do modelo. Métodos de otimização bem estabelecidos, tais como algoritmo simplex ou descida mais íngreme, não são adequados para este tipo de problemas, uma vez que seus comportamentos são fortemente dependentes de uma estimativa inicial de partida e podem ser facilmente presos num mínimo local (Press et al., 1985). Em cada iteração é gerado um novo valor dos parâmetros, até encontrar um valor de erro mínimo global durante o processo, ou quando o critério de paragem é alcançado. Neste trabalho, os algoritmos usados são o *Particle Swarm Optimization* (PSO) e o *Simulated Annealing* (SA). A utilização de aproximações estocásticas atualmente resulta numa boa propriedade de convergência global, enquanto algoritmos determinísticos são úteis para reduzir funções de estimações e tempo computacionais.

4.1. Procedimento de extração de parâmetros

A extração de parâmetros manuais para modelos semicondutores de potência, como referido no capítulo 2, é uma tarefa ainda algo complicada e alguns esforços são necessários para ajustar resultados experimentais e simulados. Uma aproximação

promissora consiste em usar métodos de extração clássicos (estimação simples fundamentado nas gamas de valores empíricas e extrapolação segundo ficha técnica dos fabricantes) para adquirir uma estimação inicial satisfatória e, em seguida, utilizar um processo de otimização de parâmetros para extrair um conjunto ótimo de parâmetros (Allard et al., 2003).

O modelo do díodo está implementado no simulador de circuitos elétricos IsSpice onde as equações diferenciais são representadas como malhas RC. Os algoritmos SA e PSO adotados são implementados em MATLAB, que tira vantagem das capacidades de matrizes do MATLAB para originar modelos compactos.

Podemos sintetizar o processo de extração de parâmetros, como se pode ver na Figura 4.1, da seguinte forma:

1. As estimativas iniciais dos parâmetros são realizadas a partir das fichas técnicas dos dispositivos e medidas básicas.
2. Medições de comportamento de comutação do dispositivo, ou seja, aquisição de formas de onda na comutação.
3. Simulação de comportamento do circuito utilizando parâmetros medidos.
4. Comparação de ondas simuladas e medidas para produzir um valor de erro.
5. Variação de valores dos parâmetros para minimizar valores do erro.

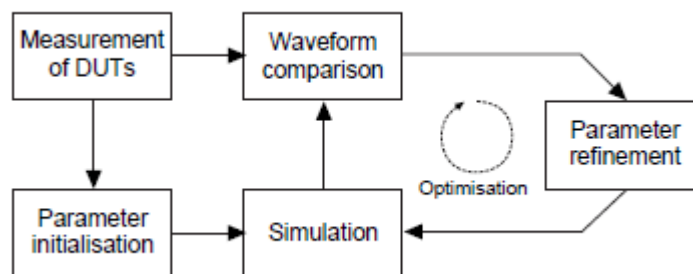


Figura 4.1 - Diagrama de procedimento de extração de parâmetros (Palmer, 2003)

No ponto 5 é realizado o procedimento de otimização. Uma vez calculado o valor do erro, os parâmetros são variados, e a simulação volta ao ponto 3 para ser executada novamente até produzir formas de ondas válidas para o conjunto de parâmetros. Por sua vez, as formas de onda simuladas são novamente comparadas com as formas de ondas medidas no ponto 4 e conseqüentemente, a otimização continua. Quando o valor dos parâmetros converge para um valor correspondente de erro mínimo, ou tenha sido alcançado o número máximo de iterações estipulado, o procedimento de otimização é terminado.

4.2. Métodos clássicos de estimação de parâmetros

Uma das principais vantagens dos modelos de díodos baseados na física é que são necessários apenas cinco parâmetros dos fabricantes e podem ser estimados a partir das fichas técnicas ou da única medida dígodo em turn-off.

As informações mais relevantes usualmente disponíveis nas fichas técnicas dos fabricantes são:

- V_{rrm} – tensão máxima repetitiva
- I_F – corrente direta DC
- Q_{rr} – carga de recuperação reversa
- I_{rm} – corrente de pico de recuperação reversa
- t_{rr} – tempo de recuperação reversa

Os passos para estimação de parâmetros são descritos como se segue.

1. A área do dígodo A : Sabendo que a densidade de corrente máxima J vária, para a maioria dos díodos de potência, entre 100 A.cm^{-2} a 150 A.cm^{-2} , a área ativa do dígodo pode ser aproximadamente estimada da corrente DC direta medida na ficha técnica como

$$A = \frac{I_F}{J} \quad (4.1)$$

Uma estimativa mais precisa pode ser obtida abrindo o dispositivo e fazendo uma medição direta (Palmer et al, 2006b).

2. Tempo de vida τ_{HL} : Se ignorar alguns efeitos de recombinação nas cargas reversa, podemos utilizar equação (4.2) para estimar τ_0 inicial para tempo de vida de nível elevado

$$\tau_{HL} = \tau_0 = \frac{Q_{rr}}{I_F} \quad (4.2)$$

No caso em que a carga de recuperação reversa Q_{rr} na ficha técnica é dado para uma temperatura diferente da temperatura ambiente, o parâmetro τ_0 pode ser adaptado de

acordo com Palmer (2003b). Segundo Lauritzen & Ma(1991), a estimação mais precisa para o tempo de vida pode ser realizada utilizando medidas do transitório turn-off.

Na Figura 4.2 podemos ver as medidas individuais utilizado para extração de parâmetros. Depois de instante T1 a forma de onda de corrente tem a seguinte formula:

$$i(t) = -I_{RM}e^{-\frac{t-T_1}{\tau_{rr}}}, t \geq T_1 \quad (4.3)$$

Onde τ_{rr} é a constante de tempo de recuperação reversa que pode ser extraído das formas de ondas de corrente de recuperação reversa experimental. Lauritzen, mostrou que a relação mantém-se como:

$$I_{mr} = a(\tau_{HL} - \tau_{rr}) \left[1 - e^{-\left(\frac{T_1}{\tau_{HL}}\right)} \right] \quad (4.4)$$

Todos os parâmetros podem ser determinados a partir das formas de onda experimentais: a inclinação da queda atual, e T1 tempo em que isto ocorre. O parâmetro τ_{HL} pode ser obtido pela resolução implícita da equação (4.4). Em alternativa, as seguintes aproximações de tempo de vida pode ser lido de um gráfico. Substituindo $I_{rm} = \alpha TA$ e $T1 = T0 + TA$. A equação (4.4) pode ser rescrita como:

$$\frac{T_A}{\tau_{rr}} = \left[\frac{\tau_{HL}}{\tau_{rr}} - 1 \right] \left\{ 1 - e^{-\left[\frac{\tau_{rr}}{\tau_{HL}} \left(\frac{T_0 + T_A}{\tau_{rr}} \right) \right]} \right\} \quad (4.5)$$

3. Largura de região deriva do díodo Wb: A dependência de campo elétrico (E) dos coeficientes de ionização para elétrons e lacunas no Si, α , exibido para ajustar os dados experimentais bastante bem nas formas de ondas (Overstraeten & Man, 1970), (R.L.Davies, 1966).

$$\alpha_{n,p} = e^{-\frac{b}{E}} \quad (4.6)$$

onde α varia entre 7×10^5 e $3.8 \times 10^6 \text{ cm}^{-1}$ com uma variação correspondido em b de $1.47 \times 10^6 \text{ Vcm}^{-1}$. Assumindo uma quebra de volume, uma junção abruta, coeficientes de ionização iguais para lacunas e elétrons e uma estrutura $p^+ - n^- - n^+$, uma expressão para tensão de ruptura em termos dos parâmetros α , b, e a largura de região (base) deriva N^-, W_B , pode ser obtido:

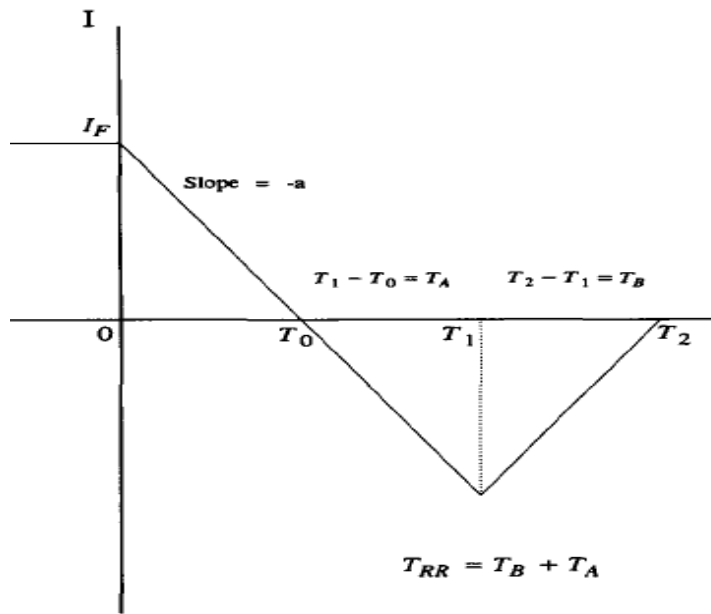


Figura 4.2 - Formas de ondas de corrente de recuperação reversa.

$$V_{BD} = \frac{bW_B}{\ln(aW_B)} \quad (4.7)$$

A largura de região (base) deriva N-, W_B , utilizado no modelo é derivado baseado na equação 4.7 utilizando $a = 1.07 \times 10^6 \text{ cm}^{-1}$, $b = 1.65 \times 10^6 \text{ V.cm}^{-1}$, e o valor da quebra de tensão de rutura retirado da ficha técnica dos fabricantes com mais alguma margem.

4. Dopagem da região deriva do díodo N_B: para fins de modelação numa gama de concentração de dopagem de impuridade efetivo ($6 \times 10^{13} - 2 \times 10^{14} \text{ cm}^{-3}$) na região deriva N-, a concentração de dopagem é assumida aproximadamente como 10^{14} cm^{-3} .

5. Parâmetros de recombinação do díodo h_p, h_n : Os parâmetros de recombinação h_p, h_n controlam a quantidade de cargas presentes no díodo durante o estado on. Os seus valores aumentam com a redução de carga armazenada e, conseqüentemente, uma redução na carga de recuperação reversa. Uma estimativa inicial de $1 \times 10^{-14} \text{ cm}^4 \cdot \text{s}^{-1}$ pode ser feita para ambos.

No modelo de extração apresentado em cima, o parâmetro tempo de vida é que domina, afetando as formas de ondas de recuperação. A primeira estimação de tempo de vida de nível

elevado, τ_{HL} é dado pelo tempo de recuperação inicial τ_0 . A estimação da largura de base N pode ser aperfeiçoada pela tentativa de melhorar o ajuste de formas de ondas do díodo. Os parâmetros de recombinação h_p e h_n afeta a corrente de recuperação reversa, em particular a corrente de recuperação reversa de pico I_{rm} e deve ser utilizado para melhorar ainda mais o parâmetro correspondente.

4.3. Estimação das formas de onda

Para estimar parâmetros por otimização há necessidade obter formas de ondas apropriados de testes de comutação indutiva para comparar com ondas simuladas. Um dos processos possíveis de comparação de sinais de ondas medidas e ondas simuladas é a utilização dos pontos salientes das formas de ondas como no caso do díodo, para a corrente de recuperação reversa e o tempo de recuperação reversa. Isto resulta num número pequeno de pontos correspondente mas também pode sofrer de baixa precisão. Mais importante, as formas de onda podem diferir de forma substancial durante os instantes de comutação, particularmente em di/dt e dv/dt . Tudo isto pode resultar num valor grande de erro na potência estimada durante a simulação. Um método de estimação muito mais preciso é simplesmente calcular a soma dos erros quadráticos por pontos de tempo nas formas de ondas. O que exige:

- Um método para fazer comparação dos diferentes traços de cada forma de onda, que normalmente são corrente e tensão.
- Normalizar erros utilizados, de modo que as exigências sujeitadas tais como fontes de tensão e correntes de cargas não afetam a consistência de extração de parâmetros.
- Sincronização precisa de formas de ondas, caso contrário os erros em di/dt e dv/dt será expressivo e induz para imprecisão de perdas de potência estimada durante a simulação.

É importante realçar que a estimação de perdas de potência é o objetivo de extração de parâmetros. Uma base possível para o erro é a dissipação de potência instantânea. Um método possível de comparar as formas de ondas é calcular a soma de erros quadráticos como foi referido acima, em cada momento de dissipação de potência instantânea. No entanto, existe uma desvantagem significativa da independência do sinal de tensão e

corrente que é perdido quando ocorrer o processo de otimização. Como exemplo simples podemos citar no caso de quando, a mesma forma de onda de dissipação de potência pode ser mantida por aumento da tensão e diminuição da corrente. Em resumo, os erros quadráticos, em vez de magnitudes de erros, atribuem mais significado a quaisquer dos grandes erros entre formas de ondas medidas e simuladas. Este é útil, uma vez que força a otimização a reduzir erros grandes primeiros e converge rapidamente. A expressão para avaliar o erro foi definida como:

$$f_{obj} = \sqrt{\sum_c \sum_i \left(\frac{(v_{exp}^i) - (v_{sim}^i)}{(v_{exp}^i)} \right)^2 + \left(\frac{(I_{exp}^i) - (I_{sim}^i)}{(I_{exp}^i)} \right)^2} \quad (4.8)$$

onde c representa o número de curvas experimentais e i o número de pontos experimentais por curva (neste trabalho, 24).

4.4. Métodos de estimação de parâmetros por otimização

As técnicas de otimização são dependentes em encontrar o erro mínimo de uma função objetivo. O conjunto ótimo de valores de parâmetros resultará no valor mínimo da função objetivo, que neste caso é calculada através da diferença entre formas de ondas simuladas e medidas. No capítulo dois, foi feita uma abordagem genérica das técnicas de otimização para engenharia. Neste capítulo, entre os vários métodos descritos no estado da arte, focaremos em duas técnicas específicas, *Particle Swarm Optimization* e *Simulated Annealing*, que são objetivo deste trabalho. Um dos grandes problemas normalmente, que um método tradicional de otimização enfrenta, é o fato de ser capaz de apenas encontrar um mínimo local e, não um mínimo global, e isto é dependente da estimação inicial. Este problema pode ser ultrapassado usando meta-heurísticas tais como *Simulated Annealing* e *Particle Swarm Optimization*, que possibilita o alcance de um mínimo mais efetivo pela introdução de variação aleatória de parâmetros (Yang et al., 2005).

Nas secções seguintes descreve-se em detalhe os dois algoritmos de otimização para o processo de extração de parâmetros analisado neste trabalho, fazendo comparação entre eles ao nível de número de interações, análise de estatísticas, erro mínimo encontrado, etc.

4.4.1. Particle Swarm Optimization para extração de parâmetros de semicondutores

O PSO, é uma das últimas técnicas de otimização evolucionária desenvolvida. É um método de otimização global heurístico e também um algoritmo de otimização baseado na inteligência de enxame.

O algoritmo é inicializado com uma população composta por partículas que são equiparadas as posições, de forma aleatória. A cada uma das partículas é facultada uma velocidade aleatória e está iterativamente em movimento através do espaço de pesquisa do problema com uma determinada dimensão. O que acontece é que as partículas são atraídas para uma determinada localização denominada melhor posição alcançada até o momento pela própria partícula e pela melhor localização alcançada em todo o enxame até o momento.

Conhecido como um algoritmo estocástico, o PSO foi inspirado pela otimização social, em que existe um processo iterativo na troca de informação das partículas vizinhas e, atualizando constantemente as suas posições. Quando surge um problema, é necessário algum meio para estimar a solução proposta e para isso, é preciso que subsista uma função fitness. O passo seguinte consiste na definição da população de indivíduos definidos como estimativas aleatórias na solução do problema inicializado. Esses indivíduos, são considerados soluções candidatas. As partículas estimam iterativamente a fitness da solução candidata e memoriza a localização onde teve mais sucesso. A melhor solução de cada partícula é chamada de melhor partícula. Os movimentos dentro do espaço de pesquisa, são conduzidos para esses sucessos, com a população normalmente a convergir, para o fim de uma experiência, numa melhor solução do problema.

O algoritmo PSO implica alguns ajuste de parâmetros que tem considerável influência no desempenho do mesmo, frequentemente designado como conflitos de interesse entre a capacidade da partícula para testar várias regiões num espaço de problema com intuito de localizar um ótimo global e a capacidade da partícula para concentrar num espaço em torno de uma promissora solução candidata para localizar precisamente um ótimo.

- A linguagem utilizada para explicar o PSO segue da analogia das partículas no enxame. Os termos chaves são (Rao, 2009) partícula: cada individuo no enxame;
- posição: coordenadas n-dimensional que representa uma solução para o problema;

- geração: número de iterações de procedimento de otimização utilizando PSO;
- *fitness*: função fitness fornece a interface entre o problema físico e o problema de otimização;
- enxame: conjunto de partículas;
- *pbest*: a posição no espaço de parâmetro de melhor fitness devolvido para uma partícula específica;
- *gbest* : a posição no espaço de parâmetros do melhor fitness devolvida em todo o enxame;

Semelhante aos outros algoritmos de otimização evolucionário baseado na população, o PSO é inicializado com uma população de partículas aleatórias utilizando uma distribuição uniforme. No entanto, cada partícula no PSO delinea uma trajetória num espaço de pesquisa n-dimensional, atualizando vetor velocidades assente nas melhores soluções encontradas até o momento por esta partícula, bem como outras no enxame.

As partículas, individualmente mantêm o controlo das suas coordenadas no espaço de problemas, que são associados com a melhor solução (fitness) que tem alcançado até ao momento, *pbest*. O valor de *gbest*, que é conduzido pela versão global do PSO é o melhor valor de todos encontrado e, a sua localização é adquirida por alguma partícula no enxame. Em cada iteração, o algoritmo PSO atualiza a velocidade de cada partícula movendo para a sua localização *pbest* e *gbest*. O procedimento para implementação é feito através dos seguintes passos (Eberhart & Shi, 1998b):

1. inicializar uma população (enxame) de partículas com posições aleatórias e velocidades no espaço de problema n-dimensional utilizando uma função de distribuição de probabilidade uniforme;
2. para cada partícula no enxame, estimar o seu valor *fitness*;
3. comparar cada estimacão de *fitness* partícula com a *pbest* da partícula atual. Se o valor atual é melhor que *pbest*, define o seu valor *pbest* para o valor atual e a localização de *pbest* para localização atual no espaço n-dimensional.

4. comparar a estimaco *fitness* com melhor conjunto da populao da iterao anterior. Se o valor atual   melhor que *gbest*, ento redefine com valor e  ndice do vetor da part cula.
5. Alterar o valor da velocidade e posio da part cula de acordo com a equao (4.9) e (4.10):

$$v_i(it + 1) = Wv_i(it) + C_1 \cdot rand. (p_{best}(it) - enxame(it)) + C_2 \cdot rand. (g_{best}(it) - enxame(it)) \quad (4.9)$$

$$x_i(it + 1) = x_i(it) + v_i(it + 1) \quad (4.10)$$

onde $i = 1, 2, \dots, N$   o  ndice da part cula, it   a iterao. As vari veis C_1 e C_2 so duas constantes positivas chamadas taxa de aprendizagem cognitiva e taxa de aprendizagem social, respetivamente, $rand$ so n meros aleat rios gerados e uniformemente distribu do no intervalo $[0, 1]$ de acordo com distribu o de probabilidade uniforme e W   o fator do peso da in rcia.

6. *Loop* para o passo 2) at  encontrar o crit rio de stop, normalmente o n mero m ximo de iteraoes.

A equao (4.9) indica que a velocidade da part cula   modificada de acordo com tr s componentes. A primeira componente   a sua velocidade anterior, $v_i(it)$, reduzido pela in rcia, W . Este componente   frequentemente conhecido como comportamento habitual. A segunda componente   uma atrao linear para sua melhor posio anterior, $p_{best}(it)$, reduzido pelo produto de C_1 e um n mero aleat rio. Note-se que um n mero aleat rio diferente   atribu do para cada dimenso. Este componente   conhecido como mem ria ou autoconhecimento. O terceiro e  ltimo componente   uma atrao linear para a melhor posio global, $g_{best}(it)$, reduzido pelo produto de C_2 , e um n mero aleat rio. Este componente   frequentemente conhecido como o “trabalho da equipa”, ou conhecimento social. Na Figura 4.3 podemos ver um mecanismo de pesquisa de uma t cnica PSO utilizando a regra de atualizao de velocidade (4.9) e a regra de atualizao da posio (4.10). A velocidade   regida no intervalo limitado entre $[V_{min}$ e $V_{max}]$ para controlar os movimentos das part culas dentro do espao de pesquisa.

A taxa de aprendizagem cognitiva e a taxa de aprendizagem social representam os pesos dos termos de aceleração estocásticos que forçam uma partícula para $pbest$ e $gbest$, respectivamente. Valores pequenos permitem uma partícula flutuar longe de região alvo. Inversamente, valores grandes resultam num movimento abrupto de partículas direcionado a região alvo. Neste trabalho, as constantes C_1 e C_2 foram ambas definidas com 2, adotando a prática típica em Eberhart e Shi (2001).

População inicial

A população inicial é gerada aleatoriamente e consiste num conjunto de n partículas (tamanho do enxame). Aplicado este algoritmo na identificação de parâmetros de díodos de potência, no qual são extraídos cinco parâmetros do díodo, a população do enxame é inicializada aleatoriamente por essas n partículas, onde cada partícula é formada por cinco parâmetros.

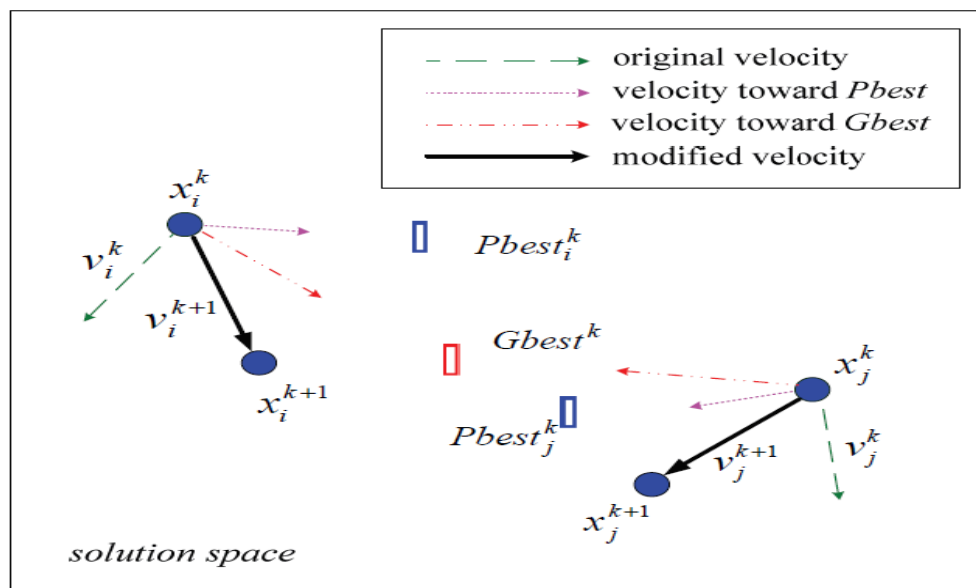


Figura 4.3 -Mecanismo de pesquisa no PSO

Enxame

Enxame é uma população aparentemente desorganizada de partículas que movem com tendência para um *cluster* juntos enquanto cada partícula parece estar a mover para direção aleatória.

4.4.1.1. Definição dos parâmetros do PSO

Tamanho da população

A partir da pesquisa anterior realizada por Shi e Eberhart (1998), está aprovado que o desempenho do algoritmo padrão não é sensível ao tamanho da população, mas à taxa de convergência. Baseado nesses resultados, o tamanho da população no presente trabalho é varia entre doze e trinta partículas a fim de manter baixos os requisitos computacionais.

Espaço de pesquisa

O intervalo em que o algoritmo calcula as variáveis de controlo ótimo é chamado de espaço de pesquisa e, o algoritmo irá procurar a melhor solução no neste espaço. Quando qualquer um dos valores de controlo ótimo de alguma partícula exceder o espaço de pesquisa, o valor será reinicializado.

Máxima iteração

Isto corresponde ao número máximo de interações permitido para convergir para uma solução ótima. Neste trabalho o número máximo de iterações é definido como 100.

Peso de inércia

A variável W , conhecida como peso da inércia, é utilizado para controlar a velocidade das partículas, assim, é responsável pelo balanceamento entre pesquisa local e global. Valores baixos de peso de inércia implica uma pesquisa local, enquanto valores altos conduz para uma pesquisa global. Inicialmente o valor do peso da inércia no algoritmo é elevado e diminui a cada iteração para um valor pequeno através da execução do PSO permitindo ao algoritmo pesquisar globalmente no início da pesquisa e, pesquisando localmente até ao fim da execução. A função do peso da inércia é dada por (Malik et al., 2007):

$$W(iter) = W_{max} - \left(\frac{W_{max} - W_{min}}{MAX_ITER} \right) \times iter; \quad (4.11)$$

A equação (4.4) mostra que o peso da inércia é atualizado de forma linear, considerando o W_{max} e W_{min} , que são peso inicial e final, respetivamente.

A interação das partículas no enxame permite a melhorar o resultado da melhor solução global da função fitness. A seleção de parâmetros, na atualização da velocidade afeta a convergência e a habilidade do enxame em encontrar o ótimo.

As equações (4.2) e (4.4) requerem a definição de alguns parâmetros de controlo. Neste trabalho adotamos valores típicos reportados na literatura (Kennedy & Eberhart, 1995c): $C_1 = 2$ e $C_2 = 2$; $W_{min} = 0.4$ e $W_{max} = 0.9$. O algoritmo foi ajustado para um tamanho da população varia entre 12 a 30 partículas e 40 a 100 iterações.

Cada partícula pesquisa o espaço de solução para a combinação que otimiza a função fitness. O valor resultante da função fitness é o atribuído da posição atual da partícula. Para a rotina de extração de parâmetros de semicondutores de potência, a função fitness a ser minimizada é o erro quadrático definido na equação (4.8).

4.4.1.2. Procedimento computacional

Esta subsecção apresenta o procedimento computacional completo do PSO proposto para minimização de funções objetivos no processo de extração de parâmetros dos díodos de potência.

Passo 1: Inicialização

- Definir $it = 1$, $i = 0$, $C1 = C2 = 2$.
- Definir iteração = 0, números dos membros da população $ENXAME_SIZE = 15$, número máximo de iterações, $MAX_ITER = 100$
- População inicial gerada aleatoriamente e o nome é $PP_i = \{PP_i, i=1,2,3,\dots, ENXAME_SIZE\}$. O valor de cada de dimensão em cada partícula é um número aleatório, uniformemente distribuído no intervalo entre $[PP_{min}, PP_{max}]$.
- Gerar a velocidade inicial e nomeá-lo como $Velocidades = Velocidades(1,i,:), i = 1,2,3,\dots, ENXAME_SIZE$. O valor de velocidade de cada partícula é um número uniforme aleatório no intervalo entre $[V_{min}, V_{max}]$.
- Utilizar a regra heurística para obter os valores dos parâmetros HP, TAU, WB, NB, A.
- Calcular função objetivo de cada partícula PP_i .
- Definir $pbest_i = PP_i$ e $fpbest$ igual a número total de todas as partículas no enxame.
- Encontrar o melhor resultado (melhor posição) entre $PP_i (i = 1,2,3,\dots, ENXAME_SIZE)$, e o resultado é colocado $fgbest$

Passo 2: Atualiza a velocidade e posição da partícula

- Atualizar a velocidade de acordo com a equação (4.9) e rand são números aleatórios uniformemente gerados entre 0 e 1.

- Atualizar a posição da partícula utilizando a equação (4.3).

- $W(ite\text{r}) = Wmax - \left(\frac{Wmax - Wmin}{MAX_ITER} \right) * ite\text{r};$

Passo 3: Atualização de pbest

- Se $f(PP_i^{it}) < fpbest_i^{it}$; $pbest_i^{it} = PP_i^{it}$ e $fpbest_i^{it} = f(PP_i^{it})$;

Passo 5: Atualizar o global best

- Para $i = 1, 2, 3, \dots, ENXAME_SIZE$, compara

$fgbest^{it}$ e $Best_EVAL(it - 1)$; se $fgbest^{it} < Best_EVAL(it - 1)$;

$Gbest^{it} = pbest_i^{it}$; $Best_EVAL(it) = fgbest^{it}$;

Passo 6: Verificar o critério de fim

- Se it não atingir MAX_ITER , volta ao passo 2, caso contrário termina o algoritmo

Como foi referido anterior, o algoritmo PSO é muito simples de implementar, exigindo apenas algumas linhas de código.

A transferência de dados entre o MATLAB e o IsSpice é o componente mais exigente na execução. Esta solução permite usufruir de toda capacidade do MATLAB, relativamente as suas rotinas associadas a acesso a ficheiros de texto e explorando as suas habilidades gráficas, permitindo por um lado acompanhar visualmente o progresso do algoritmo e, por outro lado, facilita o desenvolvimento da interface gráfica do utilizador.

Todas as funções de pré-processamento, como formatação de dados experimentais, formatação de ficheiros de simulação, ou mesmo definição da função objetivo, têm de ser inseridas no código fonte.

Afinação heurística dos parâmetros de controlo

O desafio de identificar parâmetros de controlo apropriados para o desempenho eficiente de algoritmos de otimização tem sido estudado durante muitos anos. Este problema de afinação de parâmetros para desempenhos ótimos dos algoritmos está no seu direito próprio do problema de otimização (Tweolde, Hanna, & Haskell, 2009).

Devido à falta de entendimento completo entre os parâmetros do algoritmo PSO e os seus efeitos no desempenho da otimização, a maioria dos trabalhos reportados na literatura simplesmente emprega alguns conjuntos de parâmetros genéricos para a sua aplicação. Porém, o comportamento algo confuso das partículas no algoritmo PSO e outras técnicas baseadas em enxame, é projetado para ser dependente nas definições dos diferentes parâmetros do algoritmo. Deste modo, um conjunto de parâmetros pode não ser o melhor ajuste para todos os problemas por forma a garantir resultados ótimos. Além disso, a interdependência dos diferentes parâmetros dificultam a afinação manual dos parâmetros para obter melhor desempenho. O ajuste de parâmetros constitui uma tarefa complicada no desenvolvimento deste algoritmo e a definição de espaço de pesquisa pode ser realizada de várias formas:

- Eleger um conjunto de valores em intervalos aceitáveis e com significados físicos, isto porque os algoritmos, apesar de convergirem para melhores valores, a variação desses mesmos valores podem complicar a análise estatística dos parâmetros.
- A escolha de alguns parâmetros extraídos da literatura a partir de técnicas tradicionais, divulgado na literatura, também é uma outra solução considerada interessante e que foi utilizado neste trabalho. Com a finalidade de obter resultados precisos e convergências rápidas do algoritmo, várias experiências foram realizadas.

A convergência do algoritmo também é influenciada pelo número de partículas do enxame e pela topologia: enxames fortemente ligados convergem mais rapidamente que os fracamente ligados. A melhor taxa de convergência, isto é, o compromisso entre a prospeção e a exploração, depende da função a ser otimizada, números de mínimos locais e a distância para global, a posição do ótimo global num domínio de espaço de pesquisa, a dimensão do espaço de pesquisa e a precisão requerida na localização do ótimo, etc.

Resultados experimentais reportados em (Eberhart & Shi, 1998), mostram que geralmente falando, o PSO tem uma habilidade de convergência rápida mas uma capacidade lenta de afinação dos seus parâmetros.

Centenas de simulações foram executadas de forma aleatória variando o tamanho do enxame e o número máximo de iterações. Os quatros melhores resultados apresentados pelas suas configurações das experiências foram selecionados e executados vinte vezes com valores de posições e velocidades iniciais aleatórias num intervalo $[x_{\min}, x_{\max}]$, como indicado nas tabelas mais a frente neste capítulo. Durante o processo de otimização, as partículas foram impedidas de “voar” fora do espaço de pesquisa e as velocidades foram restritas pelos intervalos definidos pelo peso da inércia. Os resultados do algoritmo PSO foram verificados utilizando outros conjuntos de amostras de dados em diferentes testes.

Antes de apresentar os resultados, no entanto, alguns critérios de convergência foram estipulados. A minimização do valor da função objetivo foi um dos critérios utilizado

pelo algoritmo PSO que pretende encontrar dentro de um número de simulações definido para 1200. O mesmo critério como um dos mais utilizados na literatura são escolhidos para facilitar a comparação direta dos resultados (James Kennedy, Russel Eberhart, 2001).

A etapa seguinte às experiências executadas para encontrar os melhores conjuntos de configurações foi calcular os valores médios, mínimos, máximos, desvio padrões e coeficiente de variação (quociente entre o desvio-padrão e a média) relativos aos parâmetros ótimos e funções objetivo para amostra de 100.

Como foi supracitado, várias simulações foram necessárias com finalidade de escolher o melhor “conjunto” de valores para os parâmetros de controlo do PSO. A escolha desses valores passa pela definição do número máximo de iterações e tamanho do enxame (nº de partículas) – Iterações / Enxame - com melhores coeficientes de variação dos seus parâmetros e menor valor da função objetivo. As Tabelas 4.2 a 4.5 mostram os resultados das simulações realizadas com diferentes dimensões do enxame e, conseqüente escolha das configurações mais adequadas para este trabalho. Pode-se facilmente observar que a Tabela 4.4 (80 / 15) apresenta melhores resultados que os restantes Tabelas. Entre todas as configurações presentes, esta apresenta menor valor da função objetivo e menor variação dos seus parâmetros, o que pode traduzir numa maior consistência dos resultados. Deste modo, suportados pelas afirmações descritas acima adotamos, esta configuração como referência para conjuntos de simulações ilustrados na Tabela 4.4.

Metodologia utilizada

A abordagem adotada neste estudo, foi uma pesquisa exaustiva para encontrar o melhor conjunto de parâmetros (A, W_B, N_B, T_{AU}, N), com o algoritmo PSO. Utilizando uma técnica basicamente empírica, os parâmetros foram explorados num intervalo definido entre um limite inferior e superior, sendo o limite superior vinte vezes maior que o limite inferior.

Testes e condições

O número de iterações requerido para alcançar os objetivos foi registado. O número máximo de simulações foi 1200, distribuídas em iterações: 40, 60, 80, 100, correspondentes a enxames de tamanho: 30, 20, 15 e 12, respetivamente. As escolhas dos parâmetros podem ter grandes impactos no desempenho de otimização

Para a análise estatística inicial foi considerada uma amostra de 20 execuções do algoritmo, permitindo reduzir o tempo de análise, uma vez que cada execução do PSO demora em média entre 20 a 30 minutos.

Tabela 4. 2 - Parâmetros extraídos (40 iterações / 30 partículas)

Parâmetros		W_B (cm)	N_B (10^{-14} cm^{-3})	A (cm^2)	T_{AU} (s)	H_P	Função Objetivo
Espaço de Pesquisa	Mínimo	0,0050	0,020	0,0200	1,0E-06	0,100	
	Máximo	0,0200	1,500	1,5000	2,0E-05	3,000	
	1	0,0160	0,291	0,1064	8,60E-06	0,9904	1,999
	2	0,0144	0,387	0,1161	1,20E-05	1,3681	1,897
	3	0,0144	0,433	0,1421	6,70E-06	1,0909	2,128
	4	0,0155	0,325	0,0835	1,40E-05	1,0151	1,894
	5	0,0129	0,612	0,1612	1,10E-05	1,6416	2,249
	6	0,0139	0,434	0,1072	1,20E-05	1,1468	2,015
	7	0,0110	1,028	0,1063	7,30E-06	0,8422	2,321
	8	0,0144	0,433	0,1421	6,70E-06	1,0909	2,128
	9	0,0159	0,340	0,0519	7,30E-06	0,3764	2,308
	10	0,0144	0,433	0,1421	6,70E-06	1,0909	2,128
	11	0,0144	0,462	0,1656	5,80E-06	1,0517	2,292
	12	0,0159	0,340	0,0519	7,30E-06	0,3764	2,308
	13	0,0144	0,433	0,1421	6,70E-06	1,0909	2,128
	14	0,0135	0,611	0,1924	6,10E-06	1,3265	2,089
	15	0,0150	0,376	0,1296	7,10E-06	1,0306	2,047
	16	0,0129	0,649	0,1549	5,60E-06	0,9814	2,113
	17	0,0113	0,972	0,2458	3,60E-06	0,8770	2,360
	18	0,0167	0,253	0,2839	5,60E-06	2,3281	1,901
	19	0,0144	0,433	0,1421	6,70E-06	1,0909	2,128
	20	0,0144	0,433	0,1421	6,70E-06	1,0909	2,128
	Mínimo	0,0110	0,2530	0,0519	0,0000	0,3764	1,8937
	Média	0,0143	0,4839	0,1405	0,0000	1,0949	2,1280
	Máximo	0,0167	1,0280	0,2839	0,0000	2,3281	2,3596
	DV	0,0015	0,2040	0,0556	0,0000	0,4059	0,1441
	CV (%)	10,7954	42,1612	39,5811	33,6855	37,0750	6,7735

Tabela 4. 3 - Parâmetros extraídos (60 Iterações / 20 partículas)

Parâmetros		W_B (cm)	N_B (10^{-14} cm^{-3})	A (cm^2)	T_{AU} (s)	H_p	Função Objetivo
Espaço de Pesquisa	Mínimo	0,0050	0,020	0,0200	1,0E-06	0,100	
	Máximo	0,0200	1,500	1,5000	2,0E-05	3,000	
	1	0,0120	0,785	0,1358	8,90E-06	1,2165	2,2807
	2	0,0132	0,492	0,1515	5,85E-06	0,9750	2,1923
	3	0,0137	0,506	0,1226	6,45E-06	0,8724	2,0913
	4	0,0143	0,147	0,0795	8,59E-06	0,9295	2,9344
	5	0,0130	0,611	0,1194	5,93E-06	0,7927	2,1194
	6	0,0155	0,440	0,0200	5,70E-06	0,1000	2,4623
	7	0,0123	0,094	0,0231	1,00E-06	0,5417	2,3432
	8	0,0120	0,785	0,1358	8,90E-06	1,2165	2,2807
	9	0,0120	0,785	0,1358	8,90E-06	1,2165	2,2807
	10	0,0132	0,492	0,1515	5,85E-06	0,9750	2,1923
	11	0,0137	0,506	0,1226	6,45E-06	0,8724	2,0913
	12	0,0120	0,785	0,1358	8,90E-06	1,2165	2,2807
	13	0,0120	0,785	0,1358	8,90E-06	1,2165	2,2807
	14	0,0132	0,492	0,1515	5,85E-06	0,9750	2,1923
	15	0,0137	0,506	0,1226	6,45E-06	0,8724	2,0913
	16	0,0143	0,147	0,0795	8,59E-06	0,9295	2,9344
	17	0,0130	0,611	0,1194	5,93E-06	0,7927	2,1194
	18	0,0155	0,440	0,0200	5,70E-06	0,1000	2,4623
	19	0,0123	0,094	0,0231	1,00E-06	0,5417	2,1212
	20	0,0130	0,601	0,1326	5,65E-06	0,8609	2,1164
	Mínimo	0,0120	0,0942	0,02	1,00E-06	0,1000	2,0913
	Média	0,0132	0,5052	0,1059	6,47E-06	0,8607	2,2934
	Máximo	0,0155	0,7850	0,1515	8,90E-06	1,2165	2,9344
	DV	0,0011	0,2324	0,0473	0,0000	0,3286	0,2468
	CV (%)	8,2818	45,9893	44,6547	35,7848	38,1839	10,7593

Tabela 4. 4 - Parâmetros extraídos (80 iterações / 15 partículas)

Parâmetros		W_B (cm)	N_B (10^{-14} cm^{-3})	A (cm^2)	T_{AU} (s)	H_P	Função Objetivo
Espaço de Pesquisa	Mínimo	0,0050	0,020	0,0200	1,0E-06	0,100	
	Máximo	0,0200	1,500	1,5000	2,0E-05	3,000	
	1	0,0141	0,401	0,2171	6,62E-06	1,8210	2,0675
	2	0,0154	0,330	0,0695	1,39E-05	0,8285	1,8885
	3	0,0136	0,449	0,1226	6,19E-06	0,9725	2,0593
	4	0,0131	0,549	0,1722	5,68E-06	1,1870	2,0735
	5	0,0141	0,401	0,2171	6,62E-06	1,8210	2,0675
	6	0,0154	0,330	0,0695	1,39E-05	0,8285	1,8885
	7	0,0136	0,449	0,1226	6,19E-06	0,9725	2,0593
	8	0,0131	0,549	0,1722	5,68E-06	1,1870	2,0735
	9	0,0141	0,401	0,2171	6,62E-06	1,8210	2,0675
	10	0,0154	0,330	0,0695	1,39E-05	0,8285	1,8885
	11	0,0136	0,449	0,1226	6,19E-06	0,9725	2,0593
	12	0,0131	0,549	0,1722	5,68E-06	1,1870	2,0735
	13	0,0141	0,401	0,2171	6,62E-06	1,8210	2,0675
	14	0,0154	0,330	0,0695	1,39E-05	0,8285	1,8885
	15	0,0136	0,449	0,1226	6,19E-06	0,9725	2,0593
	16	0,0131	0,549	0,17245	6,00E-06	1,1862	2,0735
	17	0,0141	0,401	0,2171	6,62E-06	1,8210	2,0675
	18	0,0154	0,330	0,0695	1,39E-05	0,8285	1,8885
	19	0,0136	0,449	0,1226	6,19E-06	0,9725	2,0593
	20	0,0131	0,549	0,1722	5,68E-06	1,1870	2,0735
	Mínimo	0,0131	0,330	0,0695	5,68E-06	0,8285	1,8885
	Média	0,0140	0,432	0,1454	8,12E-06	1,2022	2,0222
	Máximo	0,0154	0,549	0,2171	1,39E-05	1,8210	2,0735
	DV	0,0009	0,081	0,0565	0,0000	0,3892	0,0794
	CV (%)	6,2632	18,840	38,8874	42,5361	32,3737	3,9247

Tabela 4. 5 - Parâmetros extraídos (100 iterações / 12 partículas)

Parâmetros		W_B (cm)	N_B (10^{-14} cm^{-3})	A (cm^2)	T_{AU} (s)	H_P	Função Objetivo
Espaço de Pesquisa	Mínimo	0,0050	0,020	0,0200	1,0E-06	0,100	
	Máximo	0,0200	1,500	1,5000	2,0E-05	3,000	
	1	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	2	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	3	0,0146	0,398	0,0662	1,18E-05	0,6641	2,0318
	4	0,0144	0,430	0,1509	6,59E-06	1,1225	2,0549
	5	0,0114	0,688	0,2359	1,09E-05	2,9406	2,1462
	6	0,0068	0,645	0,02	1,03E-05	0,2550	2,6059
	7	0,0143	0,468	0,2777	4,95E-06	1,6017	2,278
	8	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	9	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	10	0,0118	1,035	0,2836	3,22E-06	0,7340	2,381
	11	0,0128	0,634	0,037	5,76E-06	0,2322	2,2006
	12	0,0127	0,623	0,02	4,75E-06	0,1000	2,4197
	13	0,0141	0,402	0,2565	4,31E-06	1,6033	2,1721
	14	0,0119	0,776	0,1899	1,02E-05	1,8612	2,2586
	15	0,0158	0,346	0,02	5,65E-06	0,1000	2,4438
	16	0,0151	0,325	0,2327	7,89E-06	2,4607	1,9206
	17	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	18	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	19	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	20	0,0137	0,472	0,1735	1,09E-05	1,8066	2,0016
	Mínimo	0,0068	0,325	0,02	3,22E-06	0,1000	1,9206
	Média	0,0133	0,527	0,1589	8,68E-06	1,4064	2,1463
	Máximo	0,0158	1,035	0,2836	1,18E-05	2,9406	2,6059
	DV	0,0019	0,166	0,0842	0,0000	0,8024	0,1924
	CV (%)	14,0467	31,476	52,9678	33,4811	57,0508	8,9624

O resultado final de análises estatísticas foi realizado com uma amostra de tamanho 100, demonstrando desta forma a robustez dos resultados apresentado pelo algoritmo. Mesmo em diferentes condições de funcionamento existe uma coerência dos resultados apresentados, o que corrobora a robustez nos parâmetros extraídos do modelo.

Como pode ser visto, o critério de variação pode fornecer erro padrão mais baixo em torno da média desejada. Isto implica que a terceira solução na Tabela 4.4 seja mais robusta com relação a variabilidade de parâmetros estimados. Notar também que o valor do erro quadrático, Função objetivo (1,8885) teria um pequeno impacto, quando de facto o efeito de variabilidade de estimação do parâmetro é muito dramático. Este resultado será exibido mais à frente neste capítulo.

Os quatro cenários apresentados em cima são resultados sumarizados depois de teste e simulações exaustivas com valores experimentais diferentes de parâmetros de controlo escolhidos de forma empírica, tendo em conta que as escolhas desse tais parâmetros podem ter grandes impactos no desempenho final da otimização.

Uma dos objetivos das técnicas de extração de parâmetros de semicondutores baseado na física, inclusive apontado no estado da arte como uma das lacunas a ser preenchido, foi o compromisso entre o tempo de execução e a capacidade de processamento consumido. Respeitando este ponto de vista, o princípio adotado foi de estipular o número de simulações como fixo (1200) e variar as iterações e tamanho do enxame. O aumento da dimensão do enxame leva à diminuição do número de iterações requeridas pelo algoritmo. Obviamente, para uma iteração, o número de partículas é igual ao número de simulações do IsSpice. Além disso, se queremos reduzir o número total de simulações necessárias para encontrar uma solução, estamos a tentar diminuir o tamanho do enxame. Mas, um enxame bastante pequeno provavelmente leva mais tempo para encontrar uma solução ou mesmo não encontrar de todo.

Alguns autores como Clerc (Clerc, 2005) baseado numa experiência empírica recomenda os valores para os tamanhos de enxames num intervalo entre 20 a 30 partículas, que de facto provou ser suficiente para resolver a maioria dos problemas de testes clássicos. Mas isto, claro, não quer dizer que são valores padrões e que funcionam da mesma forma em todos os problemas como foi comprovado neste trabalho, a técnica referência é, claro, testar!

Observando as tabelas podemos ver facilmente que devido à aleatoriedade do algoritmo na inicialização dos parâmetros, que quanto maior for o número de partículas mais variados são os seus resultados. Como tinha referido nos parágrafos anteriores, foi confirmado nas simulações que uma menor quantidade de partículas no enxame requer um maior número

de iterações para permitir ao algoritmo uma pesquisa mais profunda do espaço dimensionado e alcançar o ótimo global. A utilização de poucas partículas dá-nos uma taxa de sucesso muito baixa, como podemos observar na Tabela 4.5, onde obteve o pior resultado recorrendo a método de análise estatística. Utilizando um maior número de partículas exige um número maior de simulações por iteração. O número de iterações da Tabela 4.2 é menor mas o risco de convergência prematuro para um ponto não ótimo é maior. Por outro lado, a prospeção é favorecida em relação à exploração da pesquisa. Uma outra constatação também é o facto de que o algoritmo depende muito da estimativa inicial dos seus parâmetros, como se pode verificar numa nas diferenças das variações dos resultados apresentados pelas Tabelas.

Com as explicações e resultados consistentes apresentado acima, podemos facilmente dizer que a escolha da configuração para este trabalho recai sobre o conjunto apresentado pela Tabela 4.4, que no final será comparada com uma outra técnica de otimização promissora apresentada na próxima secção.

4.4.1.3. Resultados de simulações

Nesta secção apresenta-se os resultados de simulações obtidos pelo algoritmo PSO nas extrações de parâmetros semicondutores de potência, com atenção especial ao comportamento do algoritmo durante o processo de otimização.

Modelo dos parâmetros para os dispositivos tem sido obtido a partir de medidas de recuperação reversa, utilizando o algoritmo descrito a cima.

A Figura 4.4 ilustra o comportamento do díodo através das formas de ondas para tensão e corrente (multiplicada por um fator de 10) durante a recuperação reversa. O modelo desenvolvido demonstra uma boa aceitação de medidas experimentais e simulados. Como foi relatado na secção anterior, os algoritmos estocásticos são capazes de num espaço definido, encontrar os pontos ótimos. A Figura 4.5 mostra a evolução da função objetivo no processo de otimização e a Figura 4.6 a evolução dos parâmetros (WB - a, NB - b) do dispositivo em função da iteração. Na Figura 4.5 pode-se observar que o PSO converge para uma solução ótima rapidamente e a partir de 40 iterações mantém-se praticamente estático sem alterar os valores ou minimizar os erros quadráticos.

Na Figura 4.6 podemos observar que a amplitude do parâmetro largura da camada de depleção decai com a iteração, ao contrário do que acontece com o parâmetro NB, que alterna entre mínimos e máximos ao longo das iterações. Isto representa um raio de pesquisa padrão de uma partícula no espaço de pesquisa. Inicialmente, as partículas explorarão uma área de pesquisa ampla mas com o decorrer do tempo as partículas voltam a pesquisar uma área menor.

No processo de extração estocástica de parâmetros existe um presumível risco de tentar compensar imprecisões dos modelos através de variações de parâmetros. Se essas imprecisões forem consideráveis, as variações nos parâmetros podem ser tais que impliquem uma perda de consistência física e, com isso, uma diminuição da gama de validade do modelo.

Embora para intervalos menores o algoritmo seja mais eficiente, o facto é que nunca falhou para ótimos locais, mesmo num intervalo mais amplo. Isto para confirmar mais uma vez que o algoritmo é dependente da estimação inicial. O algoritmo foi capaz de encontrar ótimos globais em todas as execuções para um número razoável de conjunto de parâmetros. Alguns parâmetros importantes como W_B e A também foram bem-sucedidos. Outros resultados que embora sejam aceites com alguma satisfação indicam que alguns trabalhos ainda devem ser feitos para aperfeiçoar o algoritmo.

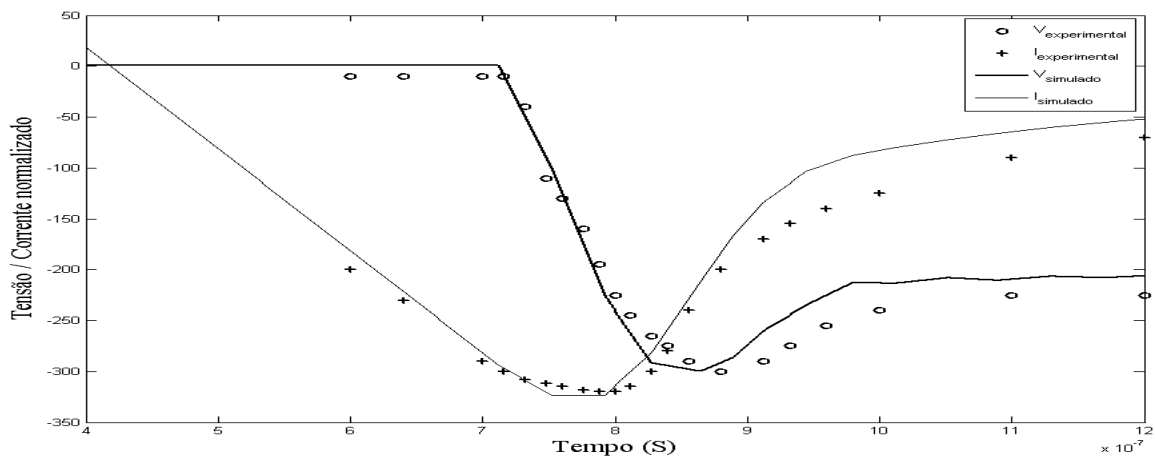


Figura 4.4 - Formas de ondas em turn-off depois de otimização do diodo – experimental e simulado

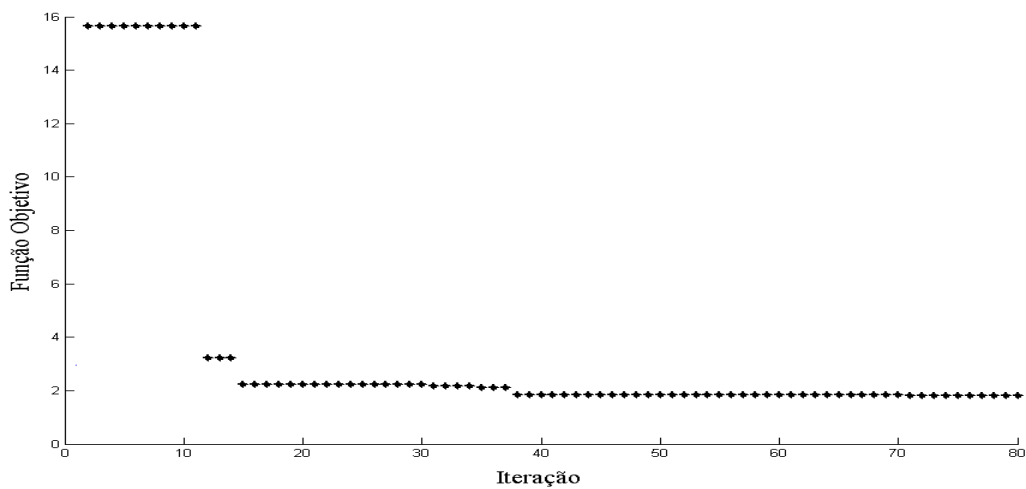


Figura 4.5 - Evolução da função com iteração

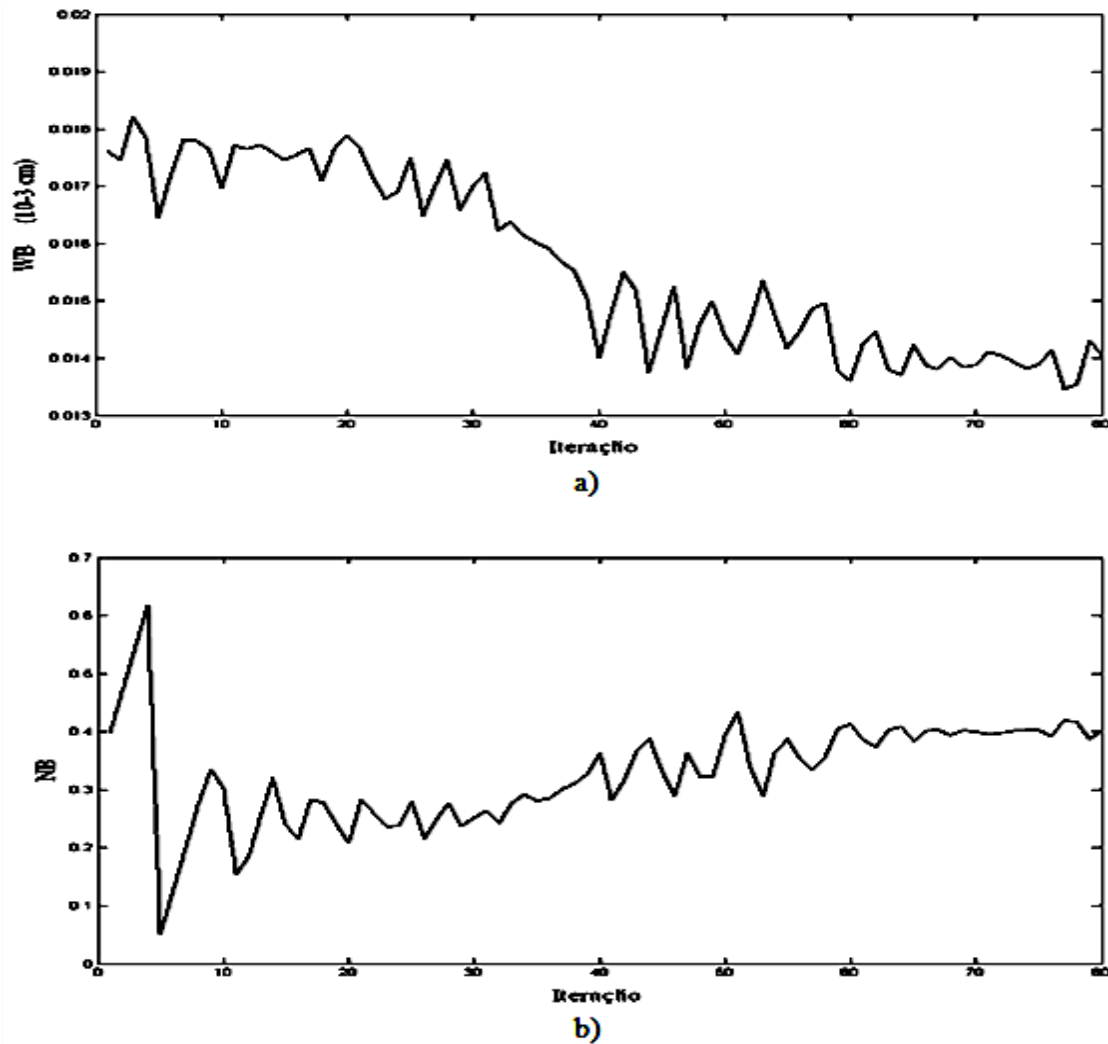


Figura 4.6 - Evolução dos parâmetros (WB-a; NB –b) com a iteração.

Por outro lado, a natureza estocástica do PSO faz com que seja difícil provar propriedades de convergência global, sabendo que um algoritmo de pesquisa local seja apenas garantido para encontrar um mínimo local de função objetivo.

Execuções com quatro de configurações de parâmetros de algoritmo PSO foram realizadas. O compromisso velocidade de convergência – robustez foi discutido.

Nas próximas secções vamos aplicar o mesmo método para o algoritmo SA e analisar o seu comportamento e no final far-se-á uma análise comparativa com o algoritmo PSO para a extração dos parâmetros de díodos semicondutores de potência.

4.4.2. Simulated Annealing para extração de parâmetros de semicondutores

Para fazer a implementação do algoritmo *Simulated Annealing* serão precisos definir alguns parâmetros de controlo como:

- Temperatura: Temperatura inicial e final desempenha um papel importante no algoritmo SA. Se a temperatura inicial escolhida for bastante pequena, o processo de pesquisa pode ser incompleto no sentido de que pode falhar a pesquisa minuciosamente do espaço de busca na localização do mínimo global antes da convergência;
- Mecanismo de perturbação: É uma técnica para criar novos vetores com valores para os parâmetros;
- Função objetivo: Uma equação escalar para calcular os erros percentuais entre os resultados experimentais e simulados;
- População inicial: Estimativa inicial dos valores dos parâmetros;
- Taxa de arrefecimento: Técnica para controlar a diminuição da temperatura. A temperatura deve ser suficientemente elevada para sair de um mínimo local e, suficientemente pequena para manter num mínimo global;
- Critério de terminação: Consiste numa técnica utilizada para controlar o fim do algoritmo. Esses critérios poderiam ser, o número de interações, erros mínimos alcançados (função objetivo) ou mesmo uma combinação entre os dois critérios;
- **Temperatura inicial**

A definição da temperatura inicial, como já foi descrito acima, recomenda-se valores altos o suficiente para obter bons resultados. Neste trabalho foram seleccionados valores entre 2.0 a 5.0 graus.

Mecanismo de perturbação

Em cada iteração foi gerado sempre um novo vetor com valores experimentais aleatórios, com médias zero e um parâmetro que depende do desvio padrão. É criado um vetor sigma que contém os valores de desvio padrão relacionado a cada parâmetro, visto que para cada parâmetro o intervalo de variação é diferente. O valor do vetor sigma está sujeito à

confiança que temos na estimativa inicial. Para alcançar um valor ótimo global, o algoritmo por vezes ignora os valores iniciais (no início da iteração) mesmo que sejam bons e foge dos mínimos locais. Os valores do vetor sigma são reduzidos iterativamente até o ser alcançado o ótimo global.

Função objetivo

Função objetivo também conhecida como função de custo, não é nada mais do que o erro percentual calculado entre dados experimentais e dados simulados. Para a otimização das formas de ondas transientes, foi definido um conjunto de parâmetros que têm uma influência enorme nas características reversas do díodo de potência. Esses parâmetros estão listados na Tabela 4.1, com os seus respectivos valores obtidos nas simulações. Para calcularmos o valor da função objetivo do algoritmo, a fórmula é a mesma definida na equação (4.1).

Além da manipulação de parâmetros de controle, o tamanho de passos no espaço de parâmetros pode ser controlado. Tem sido encontrado experimentalmente que o espaço pode ser explorado quando a taxa de aceitação de novos espaços é de 50%. Uma taxa de rejeição relativamente elevada significa que o espaço está sendo ineficientemente explorado, enquanto a taxa de rejeição relativamente alta significa que o algoritmo está a ser explorado com espaços muito pequenos.

População inicial

A estimativa inicial dos valores dos parâmetros pode ser feita de várias formas:

1. Optar por um conjunto de valores com significados físicos. Algumas vezes, é preciso refinar alguns parâmetros de controle, para que o algoritmo atinja um ótimo global, caso esteja longe do ótimo;
2. Calcular alguns parâmetros por comparação de resultados experimentais e simulados;
3. Extrair alguns parâmetros por meio de técnicas divulgadas na literatura, nas fichas técnicas, ou mesmo algumas medidas elétricas dos parâmetros;

Taxa de arrefecimento (S)

O parâmetro S desempenha um papel importante numa convergência bem sucedida do algoritmo SA. O valor para arrefecimento da temperatura, é adotada como uma taxa fixa. O valor padrão é assumido como 0,98.

Critério de terminação

O critério de terminação do algoritmo é definido como o número máximo de iterações estipulado. O número de iterações N, pode ser escolhido entre 50 a 2000, de acordo com recursos computacionais e precisão desejada para a solução.

A escolha da temperatura inicial T, do número de iterações N e a taxa de arrefecimento S, desempenham um papel importante no sucesso da convergência do algoritmo SA. Por exemplo, se por um lado, a temperatura for bastante elevada requer um número grande de iterações para convergência. Por outro lado, se a temperatura escolhida for bastante pequena, o processo de pesquisa pode ser incompleto no sentido de que possa falhar redondamente a investigação no espaço de pesquisa na localização do mínimo antes da convergência. A taxa de arrefecimento da temperatura tem um efeito similar. Valores bastante elevados de S (tais como 0.8 ou 0.9) requerem bastante esforço computacional para a convergência. Por outro lado, valores bastante pequenos de S (tais como 0.1 ou 0.2) podem resultar numa redução rápida na temperatura, o que não deve permitir uma exploração completa do espaço de pesquisa para a localização da solução mínimo global. Similarmente, um valor elevado de número máximo de iterações N, ajudará a alcançar um estado quase-equilíbrio de cada temperatura, mas resultará num maior esforço computacional. Taxas de arrefecimento mais complexas baseadas na taxa de convergência matemática esperado, tem sido utilizado na literatura para a solução de problemas de otimização prática complexa (Atiquillah & Rai, 1990). Apesar de todas as pesquisas realizadas em algoritmo SA, a escolha da temperatura inicial T, o número de iterações N, a taxa de arrefecimento S, ainda permanecem uma arte e geralmente requer um processo de tentativa e erro para encontrar valores adequados para resolver alguns tipos particulares de problemas de otimização. O procedimento de SA é mostrado com um fluxograma na figura 4.7.

De uma forma geral, executamos os seguintes passos para resolver este problema com algoritmo SA. Primeiro passo, selecionamos um conjunto de amostra, de uma população para análise. No segundo passo, utilizamos a experiência computacional para selecionar

níveis de partidas, o intervalo da variação e quantidade de alterações de cada parâmetro. Terceiro passo, selecionamos conjunto de bons parâmetros para cada problema no conjunto de análise utilizando métodos estatísticos de experimentos e respostas de otimização superficial. O último passo, calculamos a média dos valores dos parâmetros obtidos no terceiro passo para obter valores de parâmetros de qualidade aceitável.

O ISSPICE é chamado em cada iteração para executar uma simulação com conjunto de parâmetros atuais. Os erros são estimados pela diferença entre dados simulados e experimentais do qual resulta o cálculo a função objetivo.

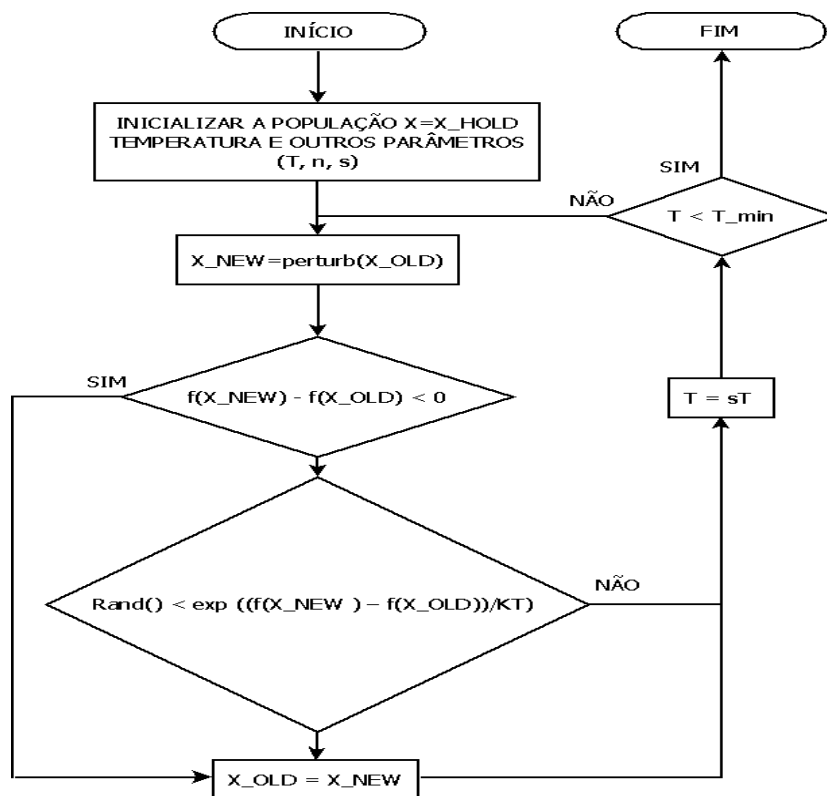


Figura 4.7 - Procedimento de Simulated Annealing

Ajuste heurístico de parâmetros

O número de iterações foi definido como 1200, igual ao número de simulações imposto no algoritmo PSO, o que facilita a comparação posterior dos resultados. Num intervalo de temperatura [0 a 10], com escalas de 0,5 foi experimentado enquanto mantendo um valor constante de escala de arrefecimento, $S = 0,98$. Abordagem semelhante à que foi adotada no processo de afinação do algoritmo PSO, foi importado para este com intuito de manter a coerência do trabalho. O processo de afinação de parâmetros SA é mais complicado do que com o PSO, sendo o facto que o SA tem mais parâmetros a serem configurados quando

comparado com outros algoritmos. Portanto, depois de várias simulações exaustivas, escolhemos as quatro configurações que indicavam melhores resultados e mais confiança e consistência dos parâmetros extraídos. Nas Tabelas 4.6 a 4.9 estão listados os resultados relativos ao processo de extração de parâmetros para díodos de potência e os valores dos erros quadráticos calculados, bem como as médias, desvio padrões e coeficiente de variação ao longo das experiências com uma amostra de 20. Os valores das temperaturas definidas são $T = 4,0, 4,5, 4,8$ e $5,0$ e a taxa de arrefecimento é fixada em $0,98$.

Uma outra consideração a levar em conta é que as medidas de desempenho devem refletir objetivos de conflitos de interesse entre uma solução de boa qualidade e com pouco tempo de consumo. A resposta que reflete o tempo para alcançar a solução é o tempo de consumo de CPU desde a iniciação de execução do algoritmo até ao tempo que o algoritmo estagnar ou parar e não para o tempo em que foi encontrado a melhor solução. Também nesta pesquisa utilizamos os cálculos da incerteza, e as variações dos resultados em percentagens. Ainda durante o processo, levamos em conta para ajuste de parâmetros, a taxa de rejeição ou aceitação com alteração do valor da temperatura durante uma execução. Segundo Kirkpatrick et al (1998), a temperatura inicial deve ser selecionada com um valor alto na fase inicial para permitir a mudança para um estado de energia melhor para pior, e assim fugir aos mínimos locais.

Cronolly (1990) obteve a temperatura inicial e final pelo uso de informações recolhidas nas sucessivas repetições iniciais. Neste método, uma série de repetições com alterações aleatórias foram feitas para alterar o valor da função objetivo. Este método é prático para diminuição da temperatura durante o processo de execução. O número inicial de repetições é 15 para determinar máximos e mínimos de função objetivo. Para determinar respostas más a temperatura deve ser reduzida. Uma forma de fazer mudança deste parâmetro é de acordo com a alteração da função objetivo ao longo das iterações.

No final desta fase encontramos os melhores conjuntos de parâmetros para algoritmo *Simulated Annealing* projetado sobre problema de extração de parâmetros de semicondutores pela resposta de metodologia de experimentos, e propusemos um procedimento baseado no método estatístico de experimento que sistematicamente seleciona parâmetros de alta qualidade.

Tabela 4. 6 - Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 4,0

Parâmetros	W _B (cm)	N _B (10 ⁺¹⁴ cm ⁻³)	A (cm ²)	T _{AU} (s)	H _P (10 ⁻¹⁴)	Função Objetivo
Valor inicial	0,0090	1,000	0,159	10E-05	1,500	
1	0,0118	0,794	0,1624	1,2E-05	1,728	2,270
2	0,0134	0,501	0,1650	1,8E-05	2,002	2,228
3	0,0173	0,224	0,1715	2,4E-05	2,683	1,780
4	0,0159	0,274	0,1542	1,5E-05	1,928	1,872
5	0,0109	0,986	0,2425	1,6E-05	2,747	2,256
6	0,0167	0,267	0,2758	1,1E-05	3,505	1,741
7	0,0098	1,449	0,1485	9,4E-06	1,324	2,360
8	0,0169	0,245	0,2095	5,6E-06	1,597	2,138
9	0,0131	0,664	0,2096	5,9E-06	1,408	2,108
10	0,0157	0,285	0,2103	2,0E-05	3,178	1,786
11	0,0122	0,693	0,1000	2,6E-05	1,252	2,252
12	0,0105	1,409	0,2621	9,2E-06	2,398	2,310
13	0,0106	1,106	0,2100	1,0E-05	2,030	2,283
14	0,0150	0,328	0,1022	1,3E-05	1,196	1,923
15	0,0231	0,115	0,1807	1,4E-05	2,572	1,766
16	0,0149	0,495	0,1944	1,2E-05	1,985	2,124
17	0,0166	0,262	0,2744	3,3E-05	4,691	1,673
18	0,0153	0,303	0,1842	1,6E-05	2,607	1,869
19	0,0095	1,858	0,2693	7,0E-06	2,167	2,343
20	0,0107	1,090	0,2767	1,7E-05	3,224	1,673
Mínimo	0,0095	0,1148	0,1000	5,6E-06	1,196	1,673
Máximo	0,0231	1,8576	0,2767	3,3E-05	4,691	2,360
Média	0,0140	0,7045	0,2072	1,4E-05	2,342	2,062
DP	0,0037	0,5452	0,0587	7,6E-06	0,989	0,245
CV (%)	26,10	77,39	28,36	54,79	42,22	11,86%

Tabela 4. 7 - Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 4,5

Parâmetros	W_B (cm)	N_B (10^{+14} cm^{-3})	A (cm^2)	T_{AU} (s)	H_P (10^{-14})	Função Objetivo
Valor inicial	0,0090	1,000	0,159	10E-05	1,500	
1	0,0092	2,202	0,1841	1,3E-05	1,834	2,390
2	0,0189	0,197	0,2727	1,3E-05	3,679	1,709
3	0,0172	0,242	0,2266	6,6E-06	2,076	1,954
4	0,0111	0,989	0,1754	1,3E-05	1,833	2,271
5	0,0112	0,906	0,1759	8,9E-06	1,613	2,282
6	0,0131	0,693	0,1001	5,7E-06	0,661	2,290
7	0,0153	0,361	0,1842	7,8E-06	1,726	1,963
8	0,0116	0,879	0,1865	2,4E-05	2,324	2,253
9	0,0109	1,066	0,1123	2,1E-05	1,293	2,280
10	0,0141	0,490	0,1277	1,2E-05	1,292	2,116
11	0,0230	0,117	0,2331	1,3E-05	3,239	1,719
12	0,0174	0,225	0,1380	3,2E-05	2,290	1,811
13	0,0106	1,147	0,1245	1,4E-05	1,314	2,295
14	0,0158	0,291	0,1595	2,4E-05	2,446	1,848
15	0,0100	1,271	0,2724	2,9E-05	3,421	2,265
16	0,0146	0,353	0,2567	8,0E-06	2,747	1,908
17	0,0118	0,764	0,1711	1,7E-05	1,986	2,247
18	0,0119	0,866	0,1756	9,1E-06	1,573	2,287
19	0,0100	1,282	0,1492	1,8E-05	1,677	2,296
20	0,0202	0,159	0,2294	9,7E-06	2,709	1,791
Mínimo	0,0100	0,1173	0,1001	5,7E-06	0,661	1,719
Máximo	0,0230	1,2821	0,2724	3,2E-05	3,421	2,296
Média	0,0140	0,6642	0,1747	1,6E-05	2,047	2,091
DV	0,0038	0,4096	0,0530	8,2E-06	0,784	0,223
CV (%)	27,32	61,66	30,35	50,70	38,29	10,65

Tabela 4. 8 - Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 4.8

Parâmetros	W _B (cm)	N _B (10 ⁺¹⁴ cm ⁻³)	A (cm ²)	T _{AU} (s)	H _P (10 ⁻¹⁴)	Função Objetivo
Valor inicial	0,0090	1,000	0,159	10E-05	1,500	
1	0,0111	1,039	0,2166	1,3E-05	2,315	2,270
2	0,0202	0,160	0,2553	1,7E-05	3,920	1,688
3	0,0168	0,314	0,1786	6,0E-06	1,163	2,291
4	0,0112	0,913	0,1006	3,2E-05	1,257	2,265
5	0,0149	0,344	0,2825	1,3E-05	3,594	1,760
6	0,0147	0,404	0,0683	1,3E-05	0,808	2,089
7	0,0119	0,786	0,2277	1,2E-05	2,427	2,250
8	0,0134	0,459	0,2509	1,2E-05	3,216	1,905
9	0,0100	1,434	0,1731	9,2E-06	1,621	2,342
10	0,0137	0,490	0,0719	1,8E-05	0,817	2,251
11	0,0127	0,679	0,2297	9,8E-06	2,223	2,254
12	0,0157	0,308	0,2073	1,4E-05	2,696	1,795
13	0,0278	0,078	0,2644	6,8E-06	2,791	1,838
14	0,0149	0,345	0,2195	1,3E-05	2,711	1,773
15	0,0160	0,263	0,1274	3,8E-05	2,102	1,855
16	0,0275	0,074	0,1448	2,7E-05	2,574	1,749
17	0,0139	0,507	0,0700	1,2E-05	0,712	2,073
18	0,0108	0,882	0,2353	2,3E-05	3,405	2,221
19	0,0192	0,179	0,1917	1,6E-05	2,770	1,756
20	0,0099	1,429	0,1976	9,6E-06	1,838	2,280
Mínimo	0,0099	0,0740	0,0683	6,0E-06	0,712	1,688
Máximo	0,0278	1,4344	0,2825	3,8E-05	3,920	2,342
Média	0,0153	0,5544	0,1857	1,6E-05	2,248	2,035
DV	0,0051	0,4080	0,0674	8,2E-06	0,952	0,238
C V(%)	33,07	73,59	36,28	52,95	42,36	11,70

Tabela 4. 9 - Sumário dos resultados de 20 experimentos e médias de desvios de respostas ótimas para T = 5.0

Parâmetros	W_B (cm)	N_B (10^{+14} cm^{-3})	A (cm^2)	T_{AU} (s)	H_P (10^{-14})	Função Objetivo
Valor inicial	0,0090	1,000	0,159	10E-05	1,500	
1	0,0137	0,488	0,1787	1,1E-05	1,911	2,007
2	0,0110	1,003	0,2288	4,7E-06	1,263	2,331
3	0,0103	1,168	0,1872	9,2E-06	1,716	2,303
4	0,0136	0,436	0,2355	1,6E-05	3,302	1,876
5	0,0122	0,538	0,1497	1,1E-05	1,843	2,230
6	0,0186	0,195	0,2440	2,6E-05	4,031	1,686
7	0,0145	0,443	0,2567	7,1E-06	2,068	2,037
8	0,0180	0,211	0,1045	1,9E-05	1,361	1,932
9	0,0104	1,195	0,1662	3,0E-05	2,060	2,273
10	0,0217	0,142	0,0541	1,2E-05	0,542	2,280
11	0,0140	0,375	0,1252	2,4E-05	1,936	2,083
12	0,0160	0,274	0,2315	1,7E-05	3,372	1,763
13	0,0110	0,933	0,1592	5,9E-06	1,076	2,328
14	0,0164	0,269	0,2121	1,9E-05	3,172	1,760
15	0,0118	0,788	0,1366	1,1E-05	1,362	2,267
16	0,0093	1,709	0,1511	1,6E-05	1,617	2,333
17	0,0243	0,100	0,1284	2,0E-05	2,046	1,808
18	0,0116	0,947	0,1919	9,4E-06	1,778	2,283
19	0,0184	0,211	0,0810	1,1E-05	0,861	2,047
20	0,0104	1,391	0,1575	1,3E-05	1,587	2,329
Mínimo	0,0093	0,1002	0,0541	4,7E-06	0,542	1,686
Máximo	0,0243	1,7094	0,2567	3,0E-05	4,031	2,333
Média	0,0144	0,6408	0,1690	1,5E-05	1,945	2,098
DV	0,0041	0,4695	0,0556	6,9E-06	0,893	0,226
CV(%)	28,76	73,27	32,89	46,75	45,90	10,80

4.4.2.1. Resultado de Simulações

A validação de extração de parâmetros neste algoritmo é semelhante ao executado pelo algoritmo PSO, através das comparações das formas de ondas simuladas e experimentais durante a recuperação reversa do diodo de potência.

A Figura 4.8 ilustra o resultado das formas de ondas obtidas durante o processo de otimização no qual culminou com boa solução. Resultado de evolução da função objetivo relativo a iteração durante a otimização é apresentado na Figura 4.9 e a Figura 4.10 mostra a evolução de alguns parâmetros do diodo (NB – a; W_B - b). Podemos observar o progresso da redução de função objetivo para a pesquisa de soluções ótimas ao longo da iteração.

Como podemos observar pelo comportamento do gráfico (Ver Figura 4.10), no início do processo de “annealing”, a temperatura é alta, o mesmo podemos dizer da função objetivo, assim a probabilidade de aceitação num movimento de subida é elevada. O percurso aleatório é capaz de explorar uma área ampla de espaço de pesquisa e evita estar preso no mínimo local. Como o parâmetro de controlo diminui, o algoritmo é focado apenas nas áreas mais promissoras. Inicialmente quando a temperatura é alta, alguns aumentos elevados de função objetivo são aceites e algumas áreas distantes de ótimos são exploradas. Como a execução continua a temperatura diminui, menos excursões de subidas são toleradas (e aqueles que são tolerados são de menor magnitude). Últimos 40% de execuções são despendidos na pesquisa em torno de ótimos. Este desempenho é típico do algoritmo SA.

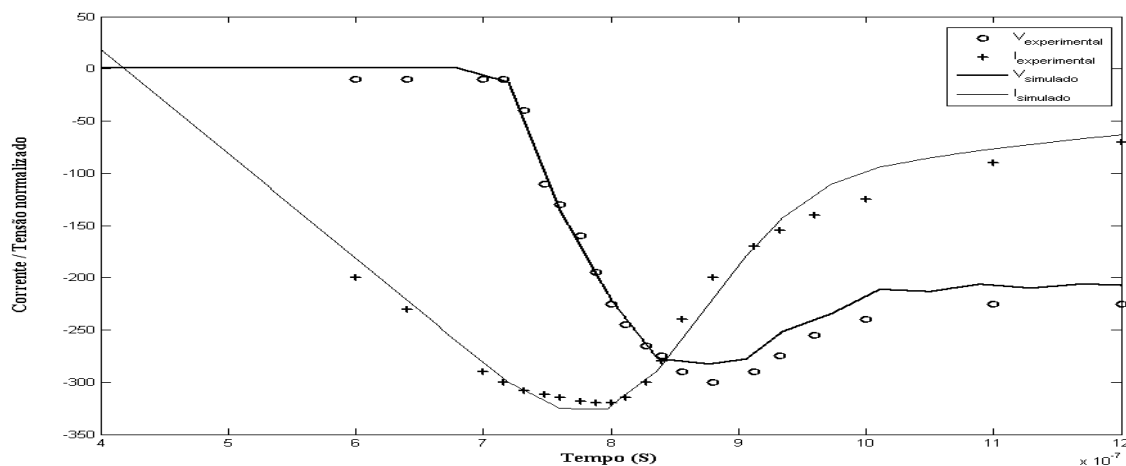
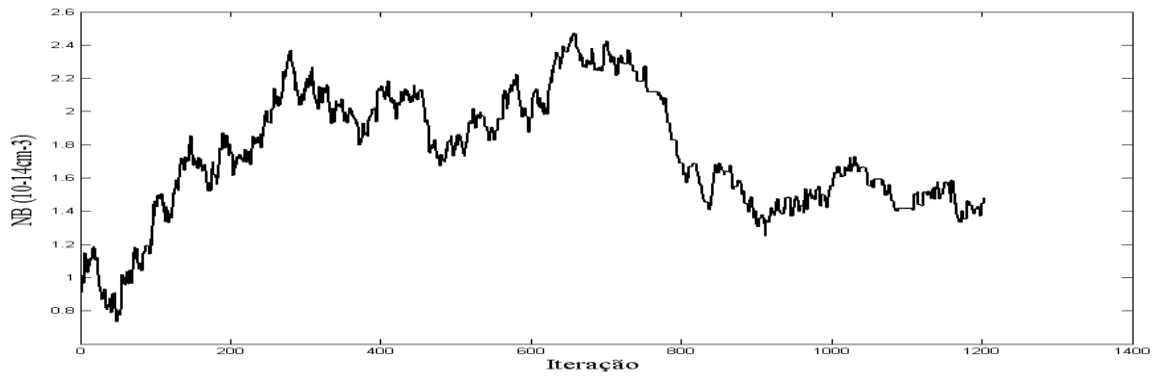
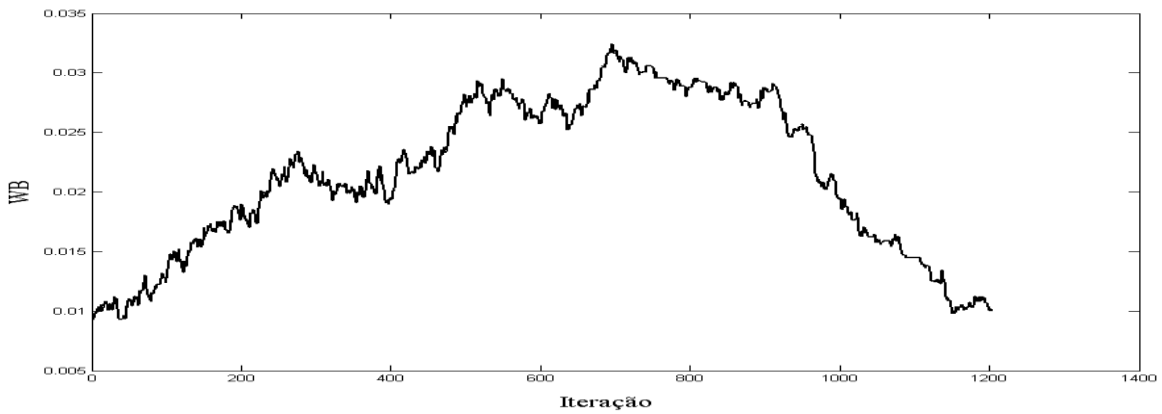


Figura 4.8 - Formas de ondas em turn-off depois da otimização do diodo – experimental e simulado



a)



b)

Figura 4.9 – Evolução dos parâmetros (NB – a; WB - b) com iteração

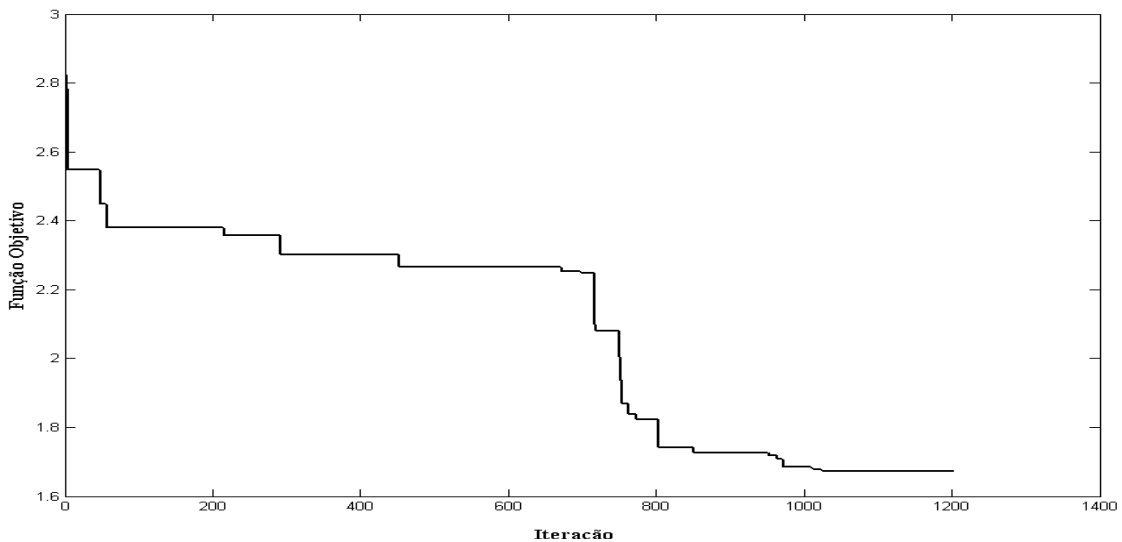


Figura 4.10 - Evolução da função objetivo com iteração

4.4.2.2. Análise comparativa

Os processos de implementação descritos nos parágrafos anteriores sobre as duas técnicas modernas de otimização para semicondutores de potência foram desenvolvidas na ferramenta MATLAB, onde foi descrita as rotinas, embutida com a ferramenta IsSpice4.

Os experimentos foram realizados na plataforma de Windows XP num desktop PC com Intel Pentium4, processador 3 GHz e 2GB de RAM. O algoritmo foi desenvolvido recorrendo a técnica de programação M file para MATLAB 2011b.

Neste trabalho, a aplicação destes dois algoritmos de pesquisa estocástica na extração de parâmetros para modelos de semicondutores de potência, nomeadamente, *Particle Swarm Optimizaton (PSO)* e *Simulated Annealing (SA)* é investigado e analisado. Os dois algoritmos são analisados e os seus méritos comparados como técnica de pesquisa alternativa quando se lida com problemas de otimização.

O método proposto PSO foi testado em comparação com algoritmo SA. Cada algoritmo foi executado 100 vezes para reduzir a aleatoriedade. A Tabela 4.10 sumariza os valores comparativos dos vários parâmetros calculados numa amostra maior com intuito de comprovar a solidez dos algoritmos.

Com intuito de comparar os algoritmos, utilizamos os mesmos conjuntos de problemas, com os mesmos cenários definidos, prevendo os pontos fracos e fortes de cada um deles.

Tabela 4. 10 - Tabela de análise estatística dos algoritmos PSO e SA com amostra 100

Parâmetros	PSO				SA			
	Mínimo	Media	DV	CV (%)	Mínimo	Media	DP	CV (%).
WB	0,0081	0,0136	0,0018	13,19%	0,0095	0,0141	0,0037	26,2%
NB	0,1686	0,4480	0,1412	31,51%	0,0777	0,6494	0,4306	66,3%
A	0,0695	0,1549	0,0535	34,53%	0,0482	0,1892	0,0508	26,9%
T _{HL}	1,8E-6	8,0E-6	3,5E-6	43,76%	5,6E-6	1,4E-5	5,7E-6	42,1%
HP	0,101	1,318	0,652	49,45%	0,4939	2,1326	0,7896	37,0%
FO ⁶	1,8212	2,1152	0,2249	10,63%	1,673	2,093	0,220	10,50%

O número máximo de iterações é frequentemente dado mas não se conhece o valor do erro médio mínimo (função objetivo) a ser alcançado. Então, um conjunto como um valor de função objetivo que é definitivamente inferior é sempre nos últimos cenários. As comparações dos algoritmos seriam feitas apenas nos melhores valores que são capazes de encontrar.

⁶ FO – Função Objetivo

A comparação será realizada segundo várias métricas. Vamos explicar como são úteis em diferentes cenários baseados nestas comparações:

- a) Precisão: Embora ambos métodos encontrem o ótimo global, a imprecisão que arrasta devido a inexatidão das equações.

No caso das abordagens baseadas na simulação, a modelação é preciso, uma vez que o SPICE é utilizado para estimar o desempenho. Contudo, o método de otimização não fornece nenhuma garantia de encontrar os mínimos globais. As técnicas *Simulated Annealing* e *Particle Swarm Optimizaton*, fornece essas garantias matemáticas. Há resultados empíricos de convergência mas não houve quaisquer estudos para testar essas convergências para o problema de otimização do circuito ou problema teste similar. Por instâncias, os algoritmos usados tiveram como referências os parâmetros de díodos de potência e erros quadráticos, onde onde tem mostrado ser capaz de encontrar ótimos globais. Possivelmente, alguns resultados de otimizações próximos de ótimos criam imprecisões em vários ótimos encontrados pela abordagem baseados na simulação.

Mesmo com esses problemas revelados acima, se por um lado, os resultados estatísticos apresentado pelo algoritmo SA é ligeiramente mais consistente de forma geral, uma vez que baseamos nos resultados do PSO para estimar valores iniciais do SA (ver Tabela 4.10). Relativamente à função objetivo, o coeficiente de variação no algoritmo PSO é de 10,63% contra 10,5% do algoritmo SA em 100 execuções para uma estimativa precisa dos resultados. Portanto, podemos constatar que a diferença não é muito significativa.

Como podemos observar os valores das taxas de variação dos parâmetros na maioria dos casos são inferiores a 50% com a exceção da taxa apresentada por um parâmetro SA que ainda assim, é inferior a 70 %.

- b) Esforço: O esforço despendido para utilizar ferramentas baseadas na simulação é moderado. Numa abordagem genérica têm de criar ficheiro IsSpice, para medições de diferentes medidas de desempenho, selecionar parâmetros a ser otimizado e definir os intervalos. Segundo algumas pesquisas, a parte que mais consome o tempo é a definição desses ficheiros IsSpice. Contudo, esse esforço pode ser reutilizado em circuitos com a mesma funcionalidade. Temos que levar em conta os esforços despendidos nas afinações dos parâmetros algoritmos ao longo deste trabalho. Numa análise geral, na codificação do algoritmo, esta etapa consome muito tempo. Embora podemos dizer que o SA apresentou melhores resultados num contexto global, seja justo também afirmar que isto deveu-se ao fato de afinação do próprio algoritmo foi baseada nos valores dos parâmetros encontrados pelo PSO. O SA com maior número de parâmetros, em que o resultado é dependente para além de uma boa estimativa inicial, dos parâmetros como temperaturas iniciais e a sua taxa de arrefecimento consome mais tempo. No caso ideal, seria necessário testar todas as combinações possíveis.

- c) Tempo: O componente tempo para dimensionamento do circuito compreende duas etapas. O primeiro, o tempo para configurar o problema de otimização. Segundo, o tempo consumido pelo algoritmo de otimização para enviar e receber valores dos parâmetros para o IsSpice. No algoritmo PSO, como antes discutido, o tempo gasto na definição do problema de otimização do circuito é elevado, o mesmo acontece para o algoritmo SA. O segundo componente de tempo, isto é, tempo de otimização é relativo ao número de parâmetros do dispositivo.

No caso como o nosso, baseado nas simulações, o tempo para configurar o problema de otimização é muito maior devido às seguintes razões: em primeiro lugar, o IsSpice é usado para estimativas em otimização e assim o tempo de otimização é limitado pelo tempo de simulação, que é muito alto para especificação transiente. Em segundo lugar, de acordo com Hellums, et al. (2000), métodos de otimização de caixa negra são métodos fracos e não específicos para propriedades (ou estruturas) de problemas de otimização. Suas vantagens é que funcionam moderadamente bem em números elevados de problemas, mas o conflito entre o seu tempo de execução é elevado. Neste trabalho, a média do tempo entre o algoritmo PSO e SA são considerados equivalentes, e em 1200 simulações varia entre 36 a 40 minutos aproximadamente, num pequeno computador desktop.

- d) Robustez: algoritmos robustos referem-se a algoritmos que garantem resultados robustos em caso de variações de parâmetros de entradas. A análise estatística foi usada para verificar a robustez dos algoritmos, medida esta muito enfatizada (Xu et al., 2005). No que diz respeito à abordagem baseada em simulações, verificar a robustez do algoritmo requer múltiplas simulações e afeta negativamente o tempo de execução do algoritmo. Apesar de vários estudos para apurar técnicas de um novo método capaz de requer um pequeno número de amostras, a precisão dessas abordagens ainda é questionável. No sentido amplo, algoritmo de otimização é útil para resolver grandes problemas rapidamente, mas necessita de elevado esforço e tempo de modelação de problema de otimização e está prometer ser robusto. Por outro lado, abordagens baseadas na simulação não são adequadas para grandes problemas (quando não decomposto), necessitam de pouco tempo e esforço na configuração e tem mostrado sucesso nos projetos robustos.

Dado o cenário apresentado, discutimos a nossa abordagem para pesquisar no campo de otimização e CAD analógica. O nosso interesse é melhorar o estado da arte na extração de parâmetros por melhorias das metodologias atuais e novas concepções. Lícitos dos compromissos numerados entre as diferentes abordagens acima, será uma contribuição se

podermos melhorar a precisão de otimização ou diminuir o seu tempo sem aumentar (ou minimamente aumentar) o esforço consumido pelos projetistas. Queremos que as nossas novas técnicas sejam adequadas para a extensão de otimização robusto.

Com este trabalho, esperamos melhorar a precisão de extração parâmetros sem aumentar o esforço despendido. Como podemos constatar através da Tabela 4.10, podemos concluir que os resultados dos erros quadráticos apresentados pelo algoritmo SA (média = 0,209) são ligeiramente melhores do que apresentado pelo algoritmo PSO (média = 0,212). O valor do desvio padrão apresentado pelos valores dos erros quadráticos indica que são aceitáveis, pois encontram-se muito próximo uns dos outros, isto, se levamos em conta que a precisão é afetada por erros acidentais. Como podemos observar no gráfico (Figura 4.5) acima que o algoritmo PSO é mais fácil para mínimos locais, e que a partir da iteração 200 não há decréscimo da função objetivo (estabiliza), enquanto o SA só estabiliza a partir das 1000 iterações (Figura 4.12). Isto demonstra que o algoritmo SA está um passo a frente e com meios mais eficazes para resolver problemas de extração de parâmetros de semicondutores de potência.

4.5. Conclusões

Neste capítulo foram propostas e analisadas duas técnicas emergentes de extração de parâmetros por meio de algoritmos de otimização. Os procedimentos baseiam-se nos algoritmos de otimização *Particle Swarm Optimizaton* e *Simulated Annealing* que por confrontação entre resultados simulados e experimentais, resulta num conjunto de parâmetros ótimos que tem como finalidade minimizar os erros entre as formas de ondas da tensão e/ou corrente experimentais e simuladas. Embora os testes experimentais detalhados tenham demonstrado que ainda falta alguma investigação, os resultados para o problema proposto são aceitáveis. Obviamente, o custo de execução extra de ajuste de parâmetros do algoritmo têm a sua cota parte nas dificuldades referidas. É nítido que a interdependência entre os diferentes parâmetros dificultam a afinação manual para melhorar o desempenho. Mas a parte interessante é que para um conjunto de parâmetros de PSO otimizado de natureza similar, funcionaram bem.

Os resultados também ilustraram que o algoritmo PSO tem habilidade para convergir rapidamente, o desempenho do PSO é sensível ao tamanho da população. Ainda sobre o PSO, podemos constatar que pode faltar capacidade para pesquisa global no fim de uma execução. Contudo, a necessidade de uma investigação para melhorar o desempenho, com alguma atenção nos ajustes de parâmetros, o método PSO pode ser uma ferramenta adequada para alcançar os objetivos propostos.

Quanto ao algoritmo SA provou ser um método geral e robusto. A sua principal vantagem notada neste trabalho é a sua flexibilidade e habilidade para abordar ótimos globais. O algoritmo é bastante versátil e não depende de quaisquer propriedades restritivas do modelo. O SA é mais facilmente ajustado que o PSO se possuir uma boa estimativa inicial.

Para qualquer sistema estocástico, um dado algoritmo de otimização pode ser ajustado para alcançar a sua performance e uma vez que leva tempo e esforço para tornar familiar com um dado código, a capacidade para ajustar um dado algoritmo para utilizar em mais que um problema deve ser considerado uma característica importante de um algoritmo.

Existe um compromisso nítido entre a qualidade de soluções e tempo requerido para calculá-lo. A precisão das estimações dos valores dos parâmetros usados na implementação do SA pode ter um efeito significativo sobre a qualidade do resultado.

Os resultados apresentados pelo algoritmo SA são mais satisfatórios, se levarmos em conta a menor variação dos seus parâmetros comparado com PSO. A modelação com PSO e SA parece altamente promissor para "portfolio" de otimização para extração de parâmetros de modelos semicondutores de potência, esperamos que a resolução dos problemas apontado no estado de arte seja cada vez mais uma certeza.

Algoritmo SA e algoritmo PSO podem coabitar juntos. O primeiro com resultados dos parâmetros ligeiramente mais consistente, e o segundo com melhores resultados dos erros quadráticos.

5. DISCUSSÃO / CONCLUSÃO

Neste trabalho a implementação, a análise e conclusões sobre algoritmos de extração de parâmetros de modelos de semicondutores baseado na física suportado por um simulador popular de circuitos elétricos foi apresentado. A técnica de modelação de elementos finitos representa uma abordagem sistemática para modelos de semicondutores de potência desenvolvida. Modelo de díodo de potência aqui fornece um desempenho superior da velocidade e precisão computacional. Assim sendo, a previsão dos diversos parâmetros e o comportamento das formas de ondas transiente durante a recuperação reversa do modelo díodo de potência e análise dos diversos modelos matemáticos, associados a algoritmos de otimização pode ser efetuada de forma automatizada com todos benefícios que lhes são intrínsecos. O compromisso entre a precisão e o tempo de convergência foi discutido. Utilizando algoritmos para teste durante turn-off do modelo díodo de potência, um conjunto ótimo de parâmetros é obtido através da maximização do desempenho do modelo com uma análise comparativa entre formas de ondas simuladas e experimentais.

A nova abordagem para modelação de díodos de potência baseada numa solução de Elementos Finitos da Equação de Difusão Ambipolar descreve completamente o comportamento não linear distribuído de portadores dentro da zona de depleção, e é facilmente implementada como um sub-circuito IsSpice que leva em conta os efeitos físicos mais importantes tais como, efeitos de recombinação do emissor, efeitos de espalhamento de portadores-portadores e a dinâmica de acumulação de cargas na zona de armazenamento.

A nova técnica de extração de parâmetros de díodos de potência baseado no algoritmo PSO foi testada e comparada com o algoritmo SA a fim de comparar os seus desempenhos, com mesma função objetivo e número de simulações.

É claro, a partir dos resultados obtidos, que o método SA supera o método PSO, na forma de onda de tensão / corrente do díodo, valor da função objetivo, e foi superado na precisão dos parâmetros e, tempo de convergência, para o modelo do díodo. Além disso, comparado com métodos gradientes convencionais, o PSO não necessita particularmente de estimativa inicial tão próxima quanto possível para as soluções, requerendo apenas um intervalo amplo específico para cada parâmetro. Assim, a técnica proposta é precisa, rápida e facilmente aplicável para extração de parâmetros de díodos de potência. Notou-se que a determinação de intervalos de pesquisa é de importância na aplicação da técnica PSO, um dos métodos de extração promissor, usado para determinar valores aproximados para parâmetros de díodos de potência. Os métodos presentes são então utilizados para melhorar a precisão dos valores extraídos, e os erros relativos em tais parâmetros como por exemplo, a largura da barreira de depleção, cujos valores têm sido determinados inicialmente com uma variação total pode ser reduzido para menos que $\pm 13\%$ com a técnica PSO proposta.

As análises dos algoritmos fornecem diretrizes qualitativas para extração de parâmetros e as implementações demonstraram serem capazes de encontrar soluções viáveis para instâncias difíceis. Com isso esperamos que os resultados representem um avanço nas relações com os resultados anteriores. Os melhores resultados foram obtidos com SA = 1,67 e PSO = 1,87; O tempo de execução para 1200 iterações foi 40 minutos. Os algoritmos apresentaram bons desempenhos e no caso do método PSO, a sua implementação é relativamente simples.

Pela observação do desempenho dos dois algoritmos em alcançar um ótimo, ou seja, a solução próxima do ótimo, conclui-se que o PSO para pesquisa local apresenta uma solução mais promissora. O algoritmo SA consome mais tempo para encontrar uma escala mas para o objetivo proposto alcançou melhores resultados que o PSO. No entanto, verificamos que, apesar de métodos de algoritmos estocásticos terem melhores desempenhos, as suas precisões são relativamente baixas. Há algumas vantagens apresentados como um tempo de execução curto mas ainda falta aumentar a sua precisão. Neste caso, fornecendo hardware adequado, esses algoritmos podem ser executados dentro de tempo muito curto.

Futuramente, pretende-se avançar com um algoritmo híbrido que combina Particle Swarm Optimization com Simulated Annealing (PSO-SA) Para identificação dos parâmetros dos modelos semicondutores de potência baseado na física. Com esta topologia híbrida do algoritmo PSO-SA tem a vantagem de por um lado, encontrar boas soluções no *Simulated Annealing* e habilidade de pesquisa rápida no *Particle Swarm Optimization*.

A sensibilidade dos parâmetros dos algoritmos estocásticos, um procedimento adequado para selecionar esses parâmetros, seguramente será de interesse da comunidade científica. Acredito, que este trabalho também sirva de referência e auxílio nos futuros projetos confrontados com a área do tema.

Referências Documentais

- [ALLEN, 1990] F. T., & Tiwari, M. K. (2007). *Swarm Intelligence – Focus on Ant and Particle Swarm Optimization*. Vienna, Austria: I-Tech Education and Publishing.
- [ARAÚJO, 2008a] Araújo, A., Chibante, R., & Carvalho, A. (2008, Novembro 13). Finite element power diode model optimized through experiment-based parameter extraction. *INTERNATIONAL JOURNAL OF NUMERICAL MODELLING: ELECTRONIC NETWORKS, DEVICES AND FIELDS*.
- [ARAÚJO, 2008b] Araújo, A., Chibante, R., & Carvalho, A. (2008, Novembro 13). Finite element power diode model optimized through experiment-based parameter extraction. *INTERNATIONAL JOURNAL OF NUMERICAL MODELLING: ELECTRONIC NETWORKS, DEVICES AND FIELDS*.
- [BAI, 2010] Bai, Q., & Tongliao. (2010, Fevereiro). *Analysis of Particle Swarm Optimization Algorithm*. *Computer and Information Science* .
- [BERTSIMAS, 1993] Bertsimas, D., & Tsitsiklis, J. (1993). *Simulated annealing*. *Statistical Science* , 8, p. 4.
- [BIRATTARI, 2001] Birattari, M., & Luis Paquete: Thomas Stutzle, K. V. (2001). *Classification of Metaheuristics and Design of Experiments for the Analysis of Components*. Darmstadt, Germany: Intellektik.
- [BRYANT, 2007] Bryant, A. T., Palmer, P. R., Santi, E., & Hudgins, J. L. (2007). *Simulink, Simulation and Optimization of Diode and Insulated Gate Bipolar Transistor Interaction in a Chopper Cell Using MATLAB and*. *IEEE* .
- [BRYANT, 2005] Bryant, A., Palmer, P., Santi, E., & Hudgins, J. (2005). *A Compact Diode Model for the Simulation of Fast Power Diodes*. *IEEE* .
- [CARVALHO, 2004] Carvalho, A., & Araújo, A. (2004., JUNHO). *A VARIATIONAL ADE (AMBIPOLAR DIFFUSION EQUATION) FORMULATION AND ITS FINITE ELEMENT SOLUTION WITH APPLICATIONS TO POWER SEMICONDUCTORS TEACHING AND SIMULATION*. *Symposium on Power Electronics, Electrical Drives, Automation and Motion, SPEEDAM* , pp. 1-3.
- [CHEN, 2005] Chen, Z. (2005). *Finite Element Methods and Their Applications*. Germany: Springer.
- [CHIBANTE, 2010] Chibante, R. (2010). *Simulated Annealing Theory with Applications*. (C. Rijeka, Ed.) *Sciyo*.
- [CLERC, 2002] Clerc, M., & Kennedy, J. (2002, Fevereiro). *The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space*. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION* , pp. 1-2.
- [CLERC, 2005] Clerc, M. (2005). *Particle Swarm Optimization*. França: Science /lavoisier.

- [CLERC, 2006] Clerc, M. (2006). Particle Swarm Optimization. London - Great Britain: ISTE Ltd.
- [COURANT, 1943] Courant, R. (1943). The finite element approximation. In R. Courant, finite element method. German.
- [Davies, 1966] R.L.Davies, R. K. (Dec. de 1966). junctions, Avalanche breakdown of diffused silicon p-n junction. IEEE Trans. Electron. Devices, Vol 13 .
- [DORIGO, 1999] Dorigo, M., & Caro, G. D. (1999). Ant Algorithms for Discrete Optimization. In P. D. Pfeifer, H. Kunz, M. M. Weber, & D. Thomas, Artificial Life. Bruxelaqs: Spring.
- [EBERHART, 1998] Eberhart, R. C., & Shi, Y. (1998). Comparison between genetic algorithms and particle swarm optimization. Evolutionary Programming VII In Evolutionary Programming VII .
- [EBERHART, 1998] Eberhart, R., & Shi, Y. (1998). A Modified Particle Swarm Optimizer. Proceedings of the IEEE International Conference on Evolutionary Computation , pp. 69-71.
- [EBERHART, 2001] Eberhart, R. C., & Shi, Y. (2001). Particle Swarm Optimization: Developments, Applications and Resources . Proc. Congress on Evolutionary Computation 2001, Seoul, Korea. Piscataway, NJ: IEEE Service Center. .
- [EBERHART, 1998] Eberhart, R. C., & Shi, Y. (1998). Comparison between Genetic Algorithms and Particle Swarm Optimization. in Proceedings of the 7th International Conference on Evolutionary Programming, .
- [ELBELTAGIA, 2005] Elbeltagia, E., Hegazyb, T., & Griersonb, D. (2005, Janeiro 19). Comparison among five evolutionary-based optimization algorithms. Science Direct: Advanced Engineering Informatics .
- [FANG, 2007] FANG, L., CHEN, P., & LIU, S. (2007, Fevereiro). Particle Swarm Optimization with Simulated Annealing for TSP. Proceedings of the 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases .
- [FATEMIZADEH, 1997]Fatemizadeh, B., Lauritzen, P. O., & Silber, D. (1997). Modeling of Power Semiconductor Devices, Problems, Limitations and Future Trends . IEEE - Univernity-Industry Research Center .
- [GLOVER, 1977] Glover, F. (1977). HEURISTICS FOR INTEGER PROGRAMMING USING SURROGATE CONSTRAINTS. Decision Sciences .
- [GLOVER, 1999] Glover, F., & Laguna, M. (1999). Tabu Search. Norwell, Massachusetts: KLUWER ACADEMIC PUBLISHERS.
- [GOEBEL, 1994] Goebel, H. (1994, Setembro). A Unified Method for Modeling Semiconductor Power Devices. IEEE TRANSACTIONS ON POWER ELECTRONICS , pp. 1-2.
- [GOEBEL, 1992] Goebel, H., & Hoffmann, K. (1992). Power diode HYBRID model with forward and reverse recovery for use in circuit simulators. IEEE .
- [GOEBEL, 1993] Goebel, H., Kraus, R., & Mattausch, H.-J. (1993). A Hybrid-Method For Modeling Semiconductor Power Devices. *SIEMENS AG, Munich - IEEE .
- [HARTMANN, 2002] Hartmann, A. K., & Rieger, H. (2002). Optimization Algorithms in Physics (Vol. 1). Berlin , Federal Republic of Germany: British Library.

- [HEFNER, 1990] Hefner, Allen R. J. (1990, Outubro). An Improved Understanding for the Transient Operation of the Power Insulated Gate (IGBT). *IEEE Trans. Power Electron*, 5.
- [HENDERSON, 2003a] Henderson, D., Jacobson, S. H., & Johnson, A. W. (2003). THE THEORY AND PRACTICE OF SIMULATED ANNEALING. In F. Glover, & G. A. Kochenberger, *HANDBOOK OF METAHEURISTICS*. Massachusetts: Kluwer Academic Publishers.
- [HENDERSON, 2003b] Henderson, D., Jacobson, S. H., & Johnson, A. W. (2003). THE THEORY AND PRACTICE OF SIMULATED ANNEALING. In F. Glover, & G. A. Kochenberger, *HANDBOOK OF METAHEURISTICS* (pp. 310-311). New York, Boston, Dordrecht, London, Moscow: Kluwer Academic Publishers.
- [HERTZ, 1995] Hertz, A., Taillard, E., & Werra, D. d. (1995). A TUTORIAL ON TABU SEARCH. Montréal, Canada H3C 3J7: EPFL, Département de Mathématiques.
- [HU, 2003] Hu, X., Eberhart, R., & Shi, Y. (2003). Engineering Optimization with Particle Swarm. *Proceedings of the IEEE Swarm Intelligence Symposium*, Indianapolis, Indiana .
- [JOSEF, 2007] Josef Dobes, M. G. (2007). Identifying Model Parameters of Semiconductor Devices Using Optimization Techniques. *Journal of Computing and Information Technology* .
- [KANG, 2003] Kang, X., E. Santi, J. H., Palmer, P., & J.F. Donlon. (2003, Fevereiro). Parameter Extraction for a Physics-Based Circuit Simulator IGBT Model. *Proc. IEEE Applied Power Electronics Conference (APEC'03)*, , pp. 946-947.
- [KENNEDY, 1995A] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proc. IEEE Int'l. Conf. on Neural IV* .
- [KENNEDY, 1995B] Kennedy, J., & Eberhart, R. (1995). A New Optimizer Using Particle Swarm Theory. *Sixth International Symposium on Micro Machine and Human Science*. IEEE Service Center Piscataway NJ .
- [KENNEDY, 2001] Kennedy, J. K. (2001). *Intelligence Swarm*. Morgan Kaufmann Publishers.
- [KIM, 1997] Kim, J.-H., & Myung, H. (1997, JULHO). Evolutionary Programming Techniques for Constrained Optimization Problems. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION* , 1.
- [KIRKPATRICK, 1983] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983, Maio 13). Optimization by Simulated Annealing. *Science, New Series* , pp. 3-5.
- [KOULAMAS, 1994] Koulamas, C., Antony, S., & Jaen, R. (1994, Janeiro). A survey of simulated annealing applications to operations-research problems. *Elsevier in its journal Omega* .
- [KRAUS, 1993] Kraus, R., & Hoffmann, K. (1993, Maio). An analytical model of IGBTs with low emitter efficiency. *IEEE: Power Semiconductor Devices and ICs, Proceedings of the 5th International Symposium on* .
- [KRAUS, 1998] Kraus, R., & Mattausch, H. J. (1998, MAIO). Status and Trends of Power Semiconductor Device Models for Circuit Simulation. *IEEE TRANSACTIONS ON POWER ELECTRONICS VOL. 13, NO. 3* , p. 14.
- [LAURITZEN, 1991] Lauritzen, P. O., & Ma, C. L. (Abril de 1991). A Simple Diode Model with Reverse Recovery. *IEEE TRANSACTIONS ON POWER ELECTRONICS* , 6.

- [LAURITZEN, 1997] Lauritzen, P. O., Sigg, J., & Ma, C. L. (1997, MAY). Modeling of Power Diodos with the Lumped-Charge Modeling Technique. IEEE TRANSACTIONS ON POWER ELECTRONICS. VOL 12 NO.3 , p. 8.
- [LAWLER, 1985] Lawler, E. L., Lenstra, J. K., Kan, A. H., & Shmoys, D. B. (1985). The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization (Wiley Series in Discrete Mathematics & Optimization) [Hardcover]. Chichester, UK: Wiley.
- [LAZINICA, 2009] Lazinica, A. (2009). Particle Swarm Optimization. (InTech, Ed.) InTech.
- [LEE, 2008] Lee, K. Y., & El-Sharkawi, M. (2008). Modern Heuristic Optimization Techniques with Applications to Power Systems. Estados Unidos: IEEE Press.
- [LETURCQ, 1995] Leturcq, P., Berraies, O., Debrie, J. L., Gillet, P., Kallala, M. A., & and J.
- [LEVENTHAL, 2006] Leventhal, R., & Green, L. (2006). Modeling Semiconductors: For Simulating Signal, Power, and Electromagnetic Integrity. woodinville, WA -USA: Springer.
- [LINDSAY, 2003] Lindsay, K. A. (2003). Numerical Solution of Partial Differential Equations. Glasgow: KAL.
- [MASSOL, 1995] Massol, L. (1995). Bipolar semiconductor device models for computer-aided design in power electronics. in Proc. 6th Eur. Conf. Power Electron .
- [MATHEW, 2005] Mathew, T. V. (2005, Janeiro 18). Genetic Algorithm. Indian Institute of Technology Bombay .
- [MATTAUSCH, 1998] Mattausch, R. K. (1998, MAIO 3). Status and Trends of Power Semiconductor - Device Models for Circuit Simulation. IEEE TRANSACTIONS ON POWER ELECTRONICS , p. 14.
- [NEDJAH, 2006] Nedjah, N., & Mourelle, L. d. (2006). Swarm Intelligent Systems (Vol. volume 26 of Studies in Computational Intelligence). Berlin/Heidelberg/New: Springer.
- [NICHOLAS, 1953] Nicholas Metropolis¹, A. W., Rosenbluth¹, M. N., Teller¹, A. H., & Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. Journal of Chemical Physics .
- [Overstraeten, 1970] Overstraeten, R. V., & Man, H. D. (1970). Measurement of the ionization rates in diffused silicon p-n junctions.
- [PALMER, 2008] Palmer, P. R., & et al. (2008, Janeiro). Physical Modeling of Fast p-i-n Diodes With Carrier Lifetime Zoning, Part I: Device Model. IEEE TRANSACTIONS ON POWER ELECTRONICS, , 23.
- [PRESS, 1992] Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). Numerical Recipes in C - Th e Art of Scientific Computing. United States of America: Press Syndicate of the University of Cambridge.
- [PRYOR, 2010] Pryor, R. W. (2010, Outubro). “Modeling PiN Photodiodes. Pryor Knowledge Systems, Inc.
- [QUIZA, 2012] Quiza et al, . R. (2012). Finite Element in Manufacturing Processes. In R. Q. al., Hybrid Modeling and Optimization of Manufacturing. Aveiro - Portugal: Springer.
- [RAO, 2009] Rao, S. S. (2009). Engineering Optimization: Theory and Practice, Fourth Edition. United States of America, United States of America: John Wiley & Sons, Inc.

- [RITZ, 1909] Ritz, W. (1909). Über eine neue Methode zur Lösung gewisser Variationsprobleme der mathematischen Physik, [On a new method for the solution of certain variational problems of mathematical physics], für reine und angewandte Mathematik .
- [ROOSBROECK, 1950]Roosbroeck, W. V. (1950, Março 30). theory of flow of electrons and holes in germanium and other semiconductors. pp. 561-562.
- [RUSSELL, 2002] Russell, S., & Norvig, P. (2002). Artificial Intelligence: A Modern Approach (2nd Edition) (3 ed.).
- [SHI, 2001] Shi, Y., & Eberart, R. C. (2001). Particle Swarm Optimization: Developments, Applications and Resources. IEEE .
- [SHOCKLEY, 1952] Shockley, W., & Read, W. T. (1952). Statistics of the Recombinations of Holes and Electrons. Phys.Rev.
- [STUTZLE, 2003] Stutzle, T., & Dorigo, M. (2003). The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances. In G. A. Kochenberger, & F. Glover, HANDBOOK OF METAHEURISTICS (pp. 252-253). New York -Estados Unidos de América: Kluwer Academic Publishers.
- [TAN, 1999] Tan, C. M., & Tseng, K. J. (Junho de 1999). Using power diode models for circuit simulations : a comprehensive review. IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, , pp. 3-4.
- [TIWARI, 2007] Tiwari, F. T. (2007). Swarm Intelligence: Focus on Ant and Particle Swarm Optimization. Croatia: I-Tech Education and Publishing.
- [TSENG, 1997] Tseng, K., Pan, S., & Foo, C. (1997). Issues in Power Diode Modelling for Circuit Simulation. IEEE .
- [TWEOLDE, 2009] Tweolde, G. S., Hanna, D. M., & Haskell, R. E. (2009). Enhancing Performance of PSO with Automatic Parameter Tuning Technique. IEEE .
- [VOGLER, 1992] Vogler, T., & Schroder, D. (1992). A NEW AND ACCURATE CIRCUIT-MODELLING APPROACH FOR THE POWER-DIODE. IEEE , pp. 1-3.
- [VOß, 1998] Voß, S., Martello, S., Osman, I. H., & Roucairol, C. (1998). Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization [Hardcover] (1^a ed.). (S. Voß, Ed.) Springer.
- Yang, W. Y., Cao, W., Chung, T.-S., & Morris, J. (2005). APPLIED NUMERICAL METHODS USING MATLAB. New Jersey;Hoboken: John Wiley & Sons.
- [WEISE, 2007] Weise, T. (2007). Global Optimization Algorithms - Theory and Application. GNU Free Documentation License, Version 1.2 .
- [WHITLEY, 1995] Whitley, D., Beveridge, R., Graves, C., & Mathias, K. (1995). Test Driving Three 1995 Genetic Algorithms: New Test Functions and Geometric Matching (1995). Journal of Heuristics .
- [ZHENG, 2004] Zheng Jianchao, J. J. (2004). Particle swam optimization Algorithm. Science Publishing Company of Beijing .
- [ZIENKIEWICZ, 2000] Zienkiewicz, O. C., & Taylor, R. L. (2000). The Finite Element Method (5^a ed., Vol. 1). Boston: Butterworth-Heinemann.