

UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA
ELÉTRICA

APLICAÇÃO DE MÉTODOS DE PONTOS INTERIORES
NO FLUXO DE POTÊNCIA ÓTIMO NÃO-LINEAR
COM UTILIZAÇÃO DE PROCESSAMENTO DE ALTO
DESEMPENHO

Tese submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a
obtenção do grau de Doutor em Engenharia Elétrica.

EDGARDO DANIEL CASTRONUOVO

Florianópolis, Maio de 2001.

APLICAÇÃO DE MÉTODOS DE PONTOS INTERIORES NO FLUXO DE POTÊNCIA ÓTIMO NÃO-LINEAR COM UTILIZAÇÃO DE PROCESSAMENTO DE ALTO DESEMPENHO

EDGARDO DANIEL CASTRONUOVO

‘Esta Tese foi julgada adequada para obtenção do Título de Doutor em Engenharia Elétrica, Área de Concentração em Sistemas de Energia, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.’

Roberto de Souza Salgado

Roberto Salgado, Ph.D.
Orientador

[Signature]

Edson Roberto De Pieri, Dr.
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

Banca Examinadora:

Roberto de Souza Salgado

Roberto Salgado, Ph.D.
Presidente

[Signature]

Jorge M. Campagnolo, D.Sc.
Co-Orientador

[Signature]

Djalma M. Falcão, Ph.D.

José Luiz Rezende Pereira

José Luiz Rezende Pereira, Ph.D.

[Signature]

Antonio J. A. Simões Costa, Ph.D.

[Signature]

Ildemar Cassana Decker, D.Sc.

*À Marcela pelo amor e dedicação,
pela companhia na vida e o constante compartilhar.
A minha mãe e irmã, que mesmo distantes estão presentes.
A Victor e Olga, pelo apoio.
“In memoriam” da minha avo.*

AGRADECIMENTOS

Aos professores Roberto de Souza Salgado e Jorge Mário Campagnolo, pela amizade e orientação durante todo o período de realização deste trabalho.

À Universidade Federal de Santa Catarina pela oportunidade de realizar este trabalho. Ao CEPEL, pela possibilidade de completar com a prática os conhecimentos adquiridos.

A todos os professores do curso de Pós-Graduação de Engenharia Elétrica, em especial aos professores Simões Costa, Hans, Decker e Aguinaldo Silveira e Silva.

Agradeço ao Laboratório de Sistemas de Potência (LABSPOT) pela infraestrutura e apoio.

Ao Centro Nacional de Super Computação da Universidade Federal de Rio Grande do Sul, pela colaboração no suporte técnico e computacional.

A Carlos Moyano, pela amizade e ajuda neste trabalho.

Aos companheiros da pós-graduação, pelo percurso transcorrido durante todo este trabalho. Saio de aqui com saudades, mais muito feliz pelo fato de ter amigos, que me auxiliaram, me compreenderam e repartiram comigo alegrias e apreensões.

Ao CNPq pelo auxílio financeiro.

Resumo da Tese apresentada à UFSC como parte dos requisitos necessários para a obtenção do grau de Doutor em Engenharia Elétrica.

APLICAÇÃO DE MÉTODOS DE PONTOS INTERIORES NO FLUXO DE POTÊNCIA ÓTIMO NÃO-LINEAR COM UTILIZAÇÃO DE PROCESSAMENTO DE ALTO DESEMPENHO

Edgardo Daniel Castronuovo

Maio/2001

Orientador: Roberto Salgado, Ph.D.

Área de Concentração: Sistemas de Energia.

Palavras-chave: Otimização, Sistemas de Potência, Fluxo de Potência Ótimo, Método de Pontos Interiores

Número de Páginas: 162

A versão Primal-Dual do método de Pontos Interiores para programação não-linear possui a capacidade de resolver problemas de otimização não-lineares de grande porte, como os exigidos na operação dos sistemas elétricos modernos.

O algoritmo inclui na sua formulação a perturbação da equação de complementaridade das condições de otimalidade de primeira ordem. Na versão convencional do método, esta perturbação requer um parâmetro empírico.

A presente monografia estuda o referido parâmetro, expressando-o como a combinação linear de duas direções: de Centralização e Afim-Escala. A possibilidade de combinar estas direções permite a formulação de outras sete versões do método, com distintas características de convergência.

As variantes do método de Pontos Interiores são avaliadas em sistemas teste e reais. São considerados no estudo: número de iterações até a convergência, tempo de cálculo e proximidade ao caminho central.

Em geral, as versões analisadas requerem tempos de cálculo por iteração superiores ao algoritmo convencional. A vetorização da solução do sistema linear através do "caminho de fatoração" é considerada a fim de diminuir os tempos de convergência do processo de otimização.

Abstract of Thesis presented to UFSC as a partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering.

ON THE APPLICATION OF NEW VERSIONS OF INTERIOR POINT METHODS TO NONLINEAR OPTIMAL POWER FLOW WITH HIGH PERFORMANCE COMPUTATION TECHNIQUES

Edgardo Daniel Castronuovo

May/2001

Advisor: Roberto Salgado, Ph.D.

Area of Concentration: Energy Systems.

Keywords: Optimization, Power Systems, Optimal Power Flow, Interior Point Methods

Number of Pages: 162

The primal-dual version of the Interior Point method is used to solve large problems of nonlinear optimization, as is demanded in the operation of the modern electric systems.

The algorithm includes in its formulation the perturbation of the complementarity equation in the first-order optimality conditions. In the conventional version of the Interior Point method, this perturbation requests of an empiric parameter.

The present monograph studies the referred parameter, expressing it as the linear combination of two directions: Centralization and Affine-Scaling. The possibility to combine these directions allows the formulation of another seven versions of the method, with different convergence characteristics.

The variants of the Interior Point methods are evaluated in test and real systems. Number of iterations to the convergence, cpu-time and neighborhood to the central path are considered in the study.

In general, the analyzed versions request of cpu-times for iteration uppers to the conventional algorithm. The vectorization of the linear system resolution through levelwise algorithms is considered in order to decrease the times of convergence of the optimization process.

Resumen de la Tesis presentada a la UFSC como parte de los requisitos necesarios para la obtención del grado de Doctor en Ingeniería Eléctrica.

APLICACIÓN DE MÉTODOS DE PUNTOS INTERIORES AL FLUJO DE POTENCIA ÓPTIMO NO LINEAL CON LA UTILIZACIÓN DE COMPUTACIÓN VECTORIAL

Edgardo Daniel Castronuovo

Mayo/2001

Orientador: Roberto Salgado, Ph.D.

Area de Estudio: Sistemas de Energía.

Palabras clave: Optimización, Sistemas de Potencia Eléctrica, Flujo de Potencia Óptimo, Método de Puntos Interiores.

Número de Páginas: 162

La versión Primal Dual del método de Puntos Interiores para Programación No Lineal tiene la capacidad de resolver problemas de optimización no lineales de grande dimensión, como los exigidos en la operación de los sistemas eléctricos modernos.

El algoritmo introduce en su formulación la perturbación de la ecuación de complementariedad de las condiciones de primer orden. En la versión convencional del método, esta perturbación utiliza un parámetro empírico.

La presente monografía estudia este parámetro, formulándolo como la combinación lineal de dos direcciones: las de Centralización y Afin-Escala. La posibilidad de combinar estas direcciones permite a formulación de otras siete versiones del método, con distintas características de convergencia.

Las variaciones del método de Puntos Interiores son analizadas en sistemas simulados y reales. Los siguientes índices de comportamiento son considerados en el estudio: número de iteraciones hasta la convergencia, tiempo de cálculo e proximidad al camino central.

Frecuentemente, las versiones analizadas tienen tiempos de cálculo por iteración superiores al algoritmo convencional. La utilización de programación vectorial en la solución del sistema lineal, por medio del camino de factoración, es utilizada con el objetivo de disminuir los tiempos de convergencia del proceso de optimización.

SUMÁRIO

Capítulo 1: Introdução	1
Capítulo 2: Conceitos Iniciais	5
2.1. Introdução	5
2.2. Otimização Matemática	5
2.3. Antecedentes do Fluxo de Potência Ótimo	7
2.4. Formulação do Fluxo de Potência Ótimo	8
2.4.1. Variáveis do FPO	8
2.4.2. Restrições de Igualdade	9
2.4.3. Restrições de Desigualdade	9
2.4.4. Função Objetivo	10
2.5. A Função Lagrangeana	11
2.5.1. Teorema da Existência dos Multiplicadores de Lagrange	12
2.6. Dualidade	13
2.6.1. Teorema da Existência das Soluções Primal e Dual	15
2.6.2. Classificação dos Métodos de Otimização para Problemas Restritos	16
2.7. As Funções de Penalidade	17
2.8. As Condições de Otimalidade de Primeira Ordem	18
2.9. Métodos de Solução do FPO	20
2.9.1. Metodologias Baseadas em Programação Linear	20
2.9.1.1. O Método Simplex	21
2.9.1.2. Métodos Baseados no Vetor Gradiente	22
2.9.1.3. Métodos de Pontos Interiores para Programação Linear	22
2.9.2. Metodologias Baseadas em Programação Não-Linear	24
2.9.2.1. Programação Quadrática Seqüencial	24
2.9.2.2. Baseados no Vetor Gradiente	24
2.9.2.3. O Método de Newton	26

2.9.2.4. Algoritmos de Pontos Interiores para Programação Não-Linear	29
2.10. Desafios do FPO	30
2.10.1. Requisitos de Resposta Temporal	31
2.10.2. Robustez da Solução	31
2.10.3. Número de Controles Afetados na Solução	32
2.10.4. Modelagem das Variáveis Discretas	32
2.10.5. Inclusão das Transações Econômicas Bilaterais no FPO	32
2.11. Abrangência do Presente Trabalho	33
2.12. Conclusões.	33

Capítulo 3: Formulação Geral do Método de Pontos Interiores Primal-Dual para Programação Não-Linear **34**

3.1. Introdução	34
3.2. O Problema Original	35
3.3. As Equações de KKT Perturbadas	36
3.3.1. Tratamento das Restrições de Desigualdade	37
3.3.2. Tratamento das Equações de Complementaridade	37
3.3.3. Formulação das Equações de KKT Perturbadas	38
3.4. A Solução Iterativa	39
3.4.1. O Método de Newton para Solução de Equações Não-Lineares	39
3.4.2. Condições Necessárias para a Utilização do Método de Newton	40
3.4.3. Montagem do Sistema Linear Completo	41
3.4.4. Redução do Sistema Linear	41
3.4.5. Características do Sistema Linear Reduzido	43
3.4.5.1. Matriz Hessiana (H)	44
3.4.5.2. Matriz Jacobiana (g(x))	45
3.4.5.3. Vetor Independente	45
3.4.6. Determinação dos Passos e Atualização das Variáveis	45
3.5. Convergência do Processo Iterativo	46
3.6. O Ponto Inicial.	47

3.7. Algoritmo Geral dos Métodos de Pontos Interiores para PNL	48
3.8. O Parâmetro da Perturbação	49
3.8.1. O Método de Pontos Interiores Primal-Dual Convencional para PNL	50
3.9. Formulação do Método de Pontos Interiores através da Barreira Logarítmica.	51
3.10. Conclusões	55
Capítulo 4: Versões do Método de Pontos Interiores	56
4.1. Introdução	56
4.2. Os Parâmetros μ e σ	57
4.3. As Direções de Centralização e Afim-Escala	59
4.4. Interpretação Gráfica das Direções	61
4.4.1. Representação da Direção de Viabilidade das Restrições de Igualdade.	62
4.4.2. O Caminho Central	63
4.4.3. As Direções Afim-Escala e de Centralização	66
4.5. Os Algoritmos de Trajetória Central	67
4.5.1. Conceitos Básicos	67
4.5.1.1. Região de Viabilidade das Restrições de Desigualdade	67
4.5.1.2. Proximidade à Trajetória Central	68
4.5.1.3. Algoritmo Geral	69
4.5.2. Algoritmo de Direções Sucessivas	70
4.5.2.1. Análise do Algoritmo	71
4.5.3. Algoritmo de Direções Sucessivas Fortemente Centralizado	72
4.5.3.1. Análise do Algoritmo	73
4.5.4. Algoritmo de Pontos Interiores Centralizado	73
4.5.4.1. Análise do Algoritmo	74
4.5.5. Otimização na Direção de Busca	75
4.5.5.1. Análise do Algoritmo	77
4.5.6. Algoritmo Preditor-Corretor de Mehrotra	77
4.5.6.1. Análise do Algoritmo	80
4.5.7. Algoritmo Preditor-Corretor com Múltiplas Correções	80

4.5.7.1. Análise do Algoritmo	82
4.5.8. Algoritmo de Máximo Passo na Trajetória Central	82
4.5.8.1. Análise do Algoritmo	85
4.6. Conclusões	85
Capítulo 5: Vetorização do Método de Pontos Interiores	86
5.1. Introdução	86
5.2. Processamento de Alto Desempenho	86
5.3. Principais Características do Processamento Vetorial	87
5.3.1. Operações Vetorizáveis	88
5.3.2. O Tempo de Preparação da Operação	89
5.3.3. Encadeamento das Unidades Funcionais	89
5.3.4. Transferência dos Dados	91
5.3.5. A Vetorização Automática	91
5.4. Vetorização de Operações Complexas	92
5.5. Utilização do Processamento Vetorial em Sistemas de Potência	93
5.6. Procedimentos de Vetorização	95
5.7. Vetorização da Solução do Sistema Linear	97
5.7.1. Introdução	97
5.7.2. Vetorização Através dos Níveis do Caminho de Fatoração	98
5.8. O Procedimento de Ordenação	102
5.9. Conclusões	104
Capítulo 6: Resultados	105
6.1. Introdução	105
6.2. O Fluxo de Potência Ótimo	105
6.3. Características dos Sistemas Teste	106
6.4. Esparsidade dos Sistemas Lineares	107
6.5. O Método de Pontos Interiores Convencional	108
6.5.1. Resultados de Convergência e Índices de Desempenho	108
6.6. Vetorização do Algoritmo Convencional	111
6.6.1. Considerações Iniciais	111
6.6.2. Resultados da Vetorização por Iteração	112

6.7. Algoritmo de Direções Sucessivas	115
6.8. Algoritmo de Direções Sucessivas Fortemente Centralizado	119
6.9. Algoritmo de Pontos Interiores Centralizado	120
6.10. Otimização na Direção de Busca	125
6.11. Algoritmo Preditor-Corretor de Mehrotra	127
6.12. Vetorização do Algoritmo Preditor-Corretor	130
6.13. Algoritmo Preditor-Corretor com Múltiplas Correções	136
6.14. Máximo Passo na Trajetória Central	139
6.14.1. Resultados da Convergência	139
6.15. Aplicação do Algoritmo de Máximo Passo na Trajetória Central	143
6.15.1. Minimização das Perdas	143
6.16. Conclusões	145
Capítulo 7: Conclusões e Sugestões Para Futuros Trabalhos	146
7.1. Introdução	146
7.2. Conclusões do Trabalho	147
7.3. Sugestões para Futuros Trabalhos	148
Bibliografia	149

LISTA DE FIGURAS

Figura 1: Despacho Econômico - Solução das Equações da Rede	7
Figura 2: Representação do Ponto de Sela	14
Figura 3: Algoritmos de Ascendência	25
Figura 4: Sistema Linear para IEEE 30 Barras	43
Figura 5: Representação da Direção de Viabilidade das Restrições de Igualdade	63
Figura 6: Representação do Caminho Central	65
Figura 7: Direções Afim-Escala, de Centralização e de Pontos Interiores	66
Figura 8: Região de Viabilidade das Restrições de Desigualdade	68
Figura 9: Matriz Exemplo	99
Figura 10: Armazenamento dos Elementos da Matriz Exemplo	102
Figura 11: Laço Recursivo no NCF IV	103
Figura 12: Laços Não Recorrentes do NCF IV	103
Figura 13: Algoritmo Convencional - Variação do μ e Máximo Resíduo. SSB 352	110
Figura 14: Algoritmo Convencional - Distâncias ao Caminho Central, em unidades de μ . SSB 352 (escala semilogarítmica)	111
Figura 15: Algoritmo Convencional - Vetorização - Ganhos por Iteração e no Total da Convergência	112
Figura 16: Algoritmo Convencional - Vetorização. Ganhos nas rotinas de Fatoração e Substituição Direta-Inversa	114
Figura 17: Direções Sucessivas - Variação do σ . SSB 352	117
Figura 18: Direções Sucessivas - Variação da Distância ao Caminho Central. SSB 352	117
Figura 19: Direções Sucessivas - Variação da $dmpd$. SSB 352	118
Figura 20: Direções Sucessivas - Variação no Mismatch	119
Figura 21: Pontos Interiores Centralizado – Variação do σ	122
Figura 22: Pontos Interiores Centralizado – Distâncias ao Caminho Central. SSB 352	122

Figura 23: Pontos Interiores Centralizado – Distâncias ao Caminho Central em Valores Absolutos. SSB 352	123
Figura 24: Pontos Interiores Centralizado – Máximo Mistmach	124
Figura 25: Preditor-Corretor - Percentagens de Redução no Número de Iterações. SSB 352	128
Figura 26: Preditor-Corretor - Variação do σ . SSB 352	128
Figura 27: Preditor-Corretor - Variação de μ .SSB 352	129
Figura 28: Preditor-Corretor - Distâncias da Trajetória de Busca ao Caminho Central. SSB 352	129
Figura 29: Preditor-Corretor - Vetorização. Ganhos na Fatoração e Substituição Direta-Inversa	131
Figura 30: Preditor-Corretor – Vetorização. Incrementos nos Tempos por Iteração (percentagens)	133
Figura 31: Preditor-Corretor – Vetorização. Redução no Número de Iterações e nos Tempos do Processo Iterativo (percentagens)	134
Figura 32: Preditor-Corretor com Múltiplas Correções - Variação do σ	138
Figura 33: Preditor-Corretor com Múltiplas Correções - Variação da Distância	138
Figura 34: Máximo Passo na Trajetória Central - Valor do Máximo Resíduo	141
Figura 35: Máximo Passo na Trajetória Central - Valor da Função Objetivo	141
Figura 36: Máximo Passo na Trajetória Central - Valor da Perturbação	141
Figura 37: Máximo Passo na Trajetória Central - Valor do Parâmetro de Combinação das Direções	142
Figura 38: Máximo Passo na Trajetória Central – Distâncias ao Caminho Central	142
Figura 39: Máximo Passo na Trajetória Central – Mínima Perda. Distâncias ao Caminho Central	145

LISTA DE TABELAS

Tabela 1: Características dos Sistemas Testes	107
Tabela 2: Esparsidade	108
Tabela 3: Algoritmo Convencional - Iterações até a Convergência	109
Tabela 4: Algoritmo Convencional – Vetorização	112
Tabela 5: Algoritmo Convencional - Tempos na Fatoração e Substituições com Vetorização Automática e Dedicada	113
Tabela 6: Direções Sucessivas - Número de Iterações	116
Tabela 7: Direções Sucessivas Fortemente Centralizado – Número de Iterações	120
Tabela 8: Pontos Interiores Centralizado - Número de Iterações	121
Tabela 9: Otimização na Direção de Busca - Valor do θ nas iterações. Algoritmo Convencional	125
Tabela 10: Preditor-Corretor - Número de Iterações	127
Tabela 11: Preditor-Corretor – Vetorização. Tempos na Fatoração e Substituição Direta-Inversa (seg.)	131
Tabela 12: Preditor-Corretor – Vetorização. Tempos até a Convergência (seg.)	132
Tabela 13: Preditor-Corretor – Vetorização. Tempos por Iteração (seg.)	132
Tabela 14: Preditor-Corretor com Múltiplas Correções - Número de Iterações	136
Tabela 15: Máximo Passo - Tempos até Convergência (min.)	140
Tabela 16: Máximo Passo – Mínima Perda. Tempos até Convergência (seg.)	144

Capítulo 1.

Introdução

Nos últimos anos, os programas de Fluxo de Potência Ótimo (FPO) têm se tornado cada vez mais essenciais como ferramenta de análise da operação dos sistemas de potência em regime permanente. Alguns fatores têm sido fundamentais para este crescimento em importância, dentre os quais: a) o surgimento de novos e potentes recursos computacionais a um custo relativamente baixo, os quais permitem reduzir o esforço requerido na solução do problema de otimização não-linear; b) a consolidação de técnicas matemáticas para a resolução de problemas de otimização de grande porte; e, c) principalmente nos últimos tempos, uma maior consideração dos interesses econômicos na operação dos sistemas de potência devido à reestruturação do mercado de energia elétrica.

À partir de um passado relativamente recente, os sistemas elétricos tem sido submetidos a uma intensa transformação na sua estrutura de mercado. A privatização dos processos de geração, transmissão e distribuição da energia elétrica têm formado mercados em muitos países. Os integrantes deste sistema econômico interagem comprando e vendendo energia, enfatizando a influência dos componentes de preço na operação do sistema de potência. Entretanto, as redes de energia elétrica possuem características especiais que as diferenciam da produção e comercialização de outros produtos. A incapacidade de armazenamento da energia elétrica em grandes blocos e a intensa interação entre os agentes envolvidos nas transações de energia do sistema requerem ferramentas matemáticas peculiares para seu tratamento. O FPO possibilita considerar na operação do sistema os preços de venda da energia, e pela sua modelagem permite a determinação dos custos de transporte, convertendo-se numa ferramenta apta para esta nova realidade. A

análise conjunta das considerações de transmissão da energia e dos aspectos econômicos permite ao FPO uma variedade de aplicações. Os agentes não estatais do Mercado Econômico de Energia visualizam a operação do sistema procurando a maximização de seus lucros, introduzindo novos paradigmas a serem considerados no FPO. Estes problemas requerem, em geral, esforços computacionais elevados, como a análise das perdas elétricas nas transações. Por conseguinte, no presente trabalho considera-se a utilização de Processamento de Alto Desempenho na solução do FPO via método de Pontos Interiores.

A solução do FPO via método de Newton de otimização (SUN *et alli.*, 1984) representou um avanço teórico significativo na área da otimização não-linear aplicada. A modelagem das restrições de desigualdade através de conjuntos ativos e inativos e alguns problemas na convergência dificultaram a aplicabilidade do método. A versão Primal-Dual do método de Pontos Interiores para Programação Não-Linear (PNL) forneceu um impulso importante ao FPO (GRANVILLE, 1994, WU, DEBS, e MARSTEN, 1994). Métodos deste tipo combinam a facilidade conceitual no tratamento das restrições de desigualdade a um desempenho computacional satisfatório. Estas características favoreceram a difusão da metodologia, resultando em variadas aplicações práticas e diversas pesquisas teóricas.

O Método de Pontos Interiores introduz uma perturbação nas equações de complementaridade de Karush-Kuhn-Tucker, com o objetivo de melhorar a convergência do processo iterativo. Nesta perturbação aparece um parâmetro do método, para o qual é utilizado um valor empírico na versão convencional. O presente trabalho visa analisar este parâmetro e fornecer para ele uma interpretação matemática. É visto no decorrer do estudo que, para o cômputo deste parâmetro, diversos métodos de Pontos Interiores são propostos. Em função do critério adotado para este cálculo, é possível diminuir o número de iterações e melhorar as características de convergência em problemas complexos. São analisados neste trabalho oito algoritmos de Pontos Interiores. Para a implementação destas versões, formula-se um algoritmo computacional para cada opção e são fornecidos detalhes de implementação.

O método de Pontos Interiores pressupõe a resolução de um sistema de equações não-lineares para a obtenção do ponto candidato a ótimo. Esta resolução é realizada através de um processo iterativo, requerendo-se em cada iteração a solução de um sistema linear de grande dimensão e alto grau de esparsidade. Esquemas de ordenação tem sido usados a fim de manter essa esparsidade, os quais fornecem a hierarquia de precedência na

resolução das linhas/colunas do sistema linear. Neste sentido, a vetorização via "caminho" de fatoração visa encontrar processos que possam ser realizados simultaneamente nos procedimentos de fatoração e substituição direta-inversa, incluindo estes cálculos em laços vetorizáveis a serem resolvidos em computadores vetoriais. Utilizando o conceito de "pipelining", estes super-processadores reduzem os tempos computacionais na resolução de operações simples entre vetores de grande dimensão, fornecendo ganhos apreciáveis em relação à operação escalar convencional. A vetorização via "caminho" de fatoração é considerada no presente trabalho a fim de melhorar a performance computacional do método de Pontos Interiores, tanto na sua forma convencional como em outras versões.

No Capítulo 2 deste trabalho são introduzidos alguns conceitos a serem usados no desenvolvimento da pesquisa. É realizada uma revisão das metodologias antecedentes ao FPO (com destaque ao Despacho Econômico Clássico) e é formulado o problema de otimização não-linear que constitui o FPO. Alguns dos conceitos matemáticos utilizados são definidos neste capítulo, incluindo dualidade, funções de penalidade, otimalidade de primeira ordem, etc. Estes conceitos permitem a compreensão dos métodos matemáticos apresentados para a resolução do FPO, com ênfase naqueles de programação não-linear.

O Capítulo 3 mostra o estudo do método de Pontos Interiores em termos da perturbação das condições de otimalidade de primeira ordem e da transformação das restrições de desigualdade. As características principais da resolução iterativa são apresentadas, e é formalizado um algoritmo geral para o método de Pontos Interiores convencional, o qual será utilizado como base para as outras versões do método. Quando da sua proposição, o método de Pontos Interiores era formulado em função de barreiras logarítmicas. Neste trabalho, é feita uma comparação entre estas formulações, com ênfase na sua equivalência.

No Capítulo 4 formulam-se as versões do método de Pontos Interiores. A direção de busca do método é decomposta em três direções fundamentais, cuja combinação determina a trajetória iterativa do algoritmo. A forma analítica de cada direção é analisada e representada geometricamente. A forma de combinar estas direções determina as versões do método de Pontos Interiores. A versão convencional e mais outras sete versões são estudadas neste capítulo.

No Capítulo 5, a vetorização do algoritmo computacional é analisada. As principais características dos processadores vetoriais são apresentadas, destacando-se as vantagens e desvantagens da sua utilização nos sistemas lineares do método de Pontos Interiores. A

vetorização proposta é formalizada, mostrando-se um exemplo de sua aplicação na substituição direta-inversa de uma matriz de pequeno porte.

No Capítulo 6 mostram-se os resultados computacionais da aplicação de métodos alternativos de Pontos Interiores ao FPO. Também são mostrados os ganhos obtidos pela vetorização do método, em duas versões do algoritmo. Resultados numéricos são mostrados para sistemas de médio e grande porte (redes elétricas do IEEE e redes realísticas representativas do sistema Sul-Sudeste Brasileiro). Quando possível, a análise dos índices de desempenho dos algoritmos é ilustrada em forma gráfica.

No Capítulo 7, são apresentadas as conclusões e sugestões para futuros trabalhos.

Capítulo 2.

Conceitos Iniciais

2.1. Introdução

Apesar do seu reconhecimento como uma ferramenta importante na decisão da operação da rede elétrica, a dimensão do problema de FPO e as não-linearidades associadas a sua modelagem dificultam a sua utilização em tempo real. Os métodos de Pontos Interiores têm se revelado uma alternativa satisfatória para a solução do FPO.

Neste capítulo, são apresentados os conceitos básicos requeridos na seqüência desta monografia. Inicialmente, o FPO é formulado como um problema geral a ser resolvido para se obter a solução das equações da rede, considerando-se as restrições operativas e os custos de operação do sistema. Posteriormente, é feita uma introdução aos métodos de otimização utilizados para a resolução do FPO, com ênfase na modelagem não-linear.

2.2. Otimização Matemática

A otimização da operação de um sistema físico pode ser interpretada como a determinação da "melhor" solução de um problema analiticamente bem definido, o qual é suposto ser uma representação satisfatória do comportamento real do sistema. Esta área inclui o estudo dos critérios de otimalidade para os problemas, a determinação de métodos de solução, o estudo da estrutura desses métodos e a implementação computacional em problemas teste e reais.

Antes de 1940, os métodos para a otimização numérica de funções multivariáveis eram pouco conhecidos. Até esta época, apenas metodologias baseadas em direções descendentes e o método de Newton eram utilizados. Entretanto, segundo FLETCHER (1987), as aplicações desses métodos eram convenientes apenas para problemas de

pequena dimensão. O avanço da informática em termos tanto de hardware quanto de software, facilitou a aplicação das técnicas de otimização em problemas de grande porte, tendo impulsionado a pesquisa de novos métodos e a análise de suas características.

Entre os anos 1940 e 1960, observou-se a introdução e o desenvolvimento de um importante conjunto de métodos na área da Programação Linear, o qual utilizava aproximações lineares da realidade física. Posteriormente, algoritmos para sistemas de grande porte foram desenvolvidos na área de em Programação Quadrática e Não-Linear (FLETCHER, 1987).

A formulação geral de um problema de otimização consiste na minimização de um índice de desempenho, representado analiticamente por uma função, sujeito a um conjunto de equações e inequações, que representam o comportamento e as limitações físicas do sistema e que são denominadas restrições. Em termos matemáticos, este problema é expresso como

$$\text{Min. } f(x) \quad (2.1)$$

$$\text{s.a. } g(x) = 0 \quad (2.2)$$

$$h(x) \leq 0 \quad (2.3)$$

onde, $f(x)$ é o índice de desempenho ou função objetivo; $g(x)$ e $h(x)$ são as restrições de igualdade e desigualdade, respectivamente; e x são as variáveis de otimização do problema.

O sistema físico é modelado através das restrições de igualdade e desigualdade (2.2) e (2.3). O índice de desempenho ou função objetivo (2.1) representa a quantidade de especial interesse que se deseja otimizar, sendo peculiar a cada sistema físico.

As aplicações da otimização na área de engenharia são variadas. Diversos problemas de engenharia podem ser representados como a minimização (ou maximização) de alguma determinada quantidade sujeito a limites impostos pela realidade do sistema em questão. Neste trabalho, o problema analisado é o chamado Fluxo de Potência Ótimo (FPO).

2.3. Antecedentes do Fluxo de Potência Ótimo

A modelagem de um sistema de potência em regime permanente é realizada de acordo com as leis de Kirchoff. A aplicação destas leis resulta num conjunto de equações algébricas não-lineares de grande porte (do tipo $g(x) = 0$), conhecido como Fluxo de Potência ou Fluxo de Carga, o qual é resolvido através de um processo iterativo.

O Fluxo de Potência permite conhecer o estado do sistema na situação atual e modelar situações futuras numa forma relativamente simples. Entretanto, esta modelagem possui as seguintes limitações: a) necessidade da pré-especificação de variáveis, b) dificuldade na modelagem de restrições de desigualdade e c) impossibilidade de modelagem dos custos associados à operação da rede elétrica

Para se suprir uma determinada demanda, as potências ativas fornecidas pelos geradores devem ser pré-estipuladas (exceto para o gerador que "supre as perdas"). Um possível indicador para a especificação da quantidade de potência gerada por cada unidade seria o custo associado a esta geração. A adição desta consideração ao problema de distribuição de potência resultou no desenvolvimento do Despacho Econômico Clássico.

A posterior combinação do Despacho Econômico Clássico com a solução das equações da rede resultou no processo iterativo de dois passos ilustrado na Figura 1.

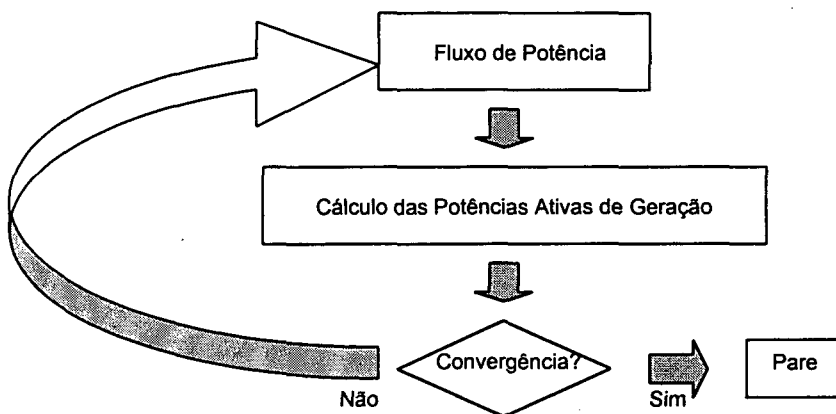


Figura 1. Despacho Econômico-Solução das Equações da Rede Elétrica

Como mostra a Figura 1, o Despacho Econômico com solução das equações da rede consiste de dois processos iterativos englobados num processo iterativo maior. O

primeiro destes resolve o problema de Fluxo de Potência. O segundo determina a solução de um problema de otimização reduzido, no qual minimiza-se o custo de geração de potência ativa. Neste segundo processo, a rede é modelada através de: a) uma equação que representa o balanço total de potência ativa; b) restrições de desigualdade que modelam os limites máximo e mínimo de geração de potência ativa. As únicas variáveis consideradas são as potências ativas de geração, resultando num problema de otimização simples e de rápida resolução, a ser resolvido através de um processo iterativo. As potências ativas calculadas neste passo são introduzidas como constantes no Fluxo de Potência, continuando-se o processo iterativo até atingir a convergência.

Atualmente, o Despacho Econômico é considerado como uma simplificação de um problema de otimização mais abrangente conhecido como Fluxo de Potência Ótimo.

2.4. Formulação do Fluxo de Potência Ótimo

A primeira formulação do FPO, atribuída a CARPENTIER (1962), era excessivamente complexa, razão pela qual não foi adotada posteriormente. Simplificações naquele modelo, resultaram numa formulação na qual resolvem-se num único problema de otimização as equações do Fluxo de Potência e a minimização de um índice de desempenho, considerando-se as limitações de operação do sistema. O FPO assume a forma geral de um problema de otimização (equações (2.1) à (2.3)), consistindo de uma função objetivo sujeita a restrições de igualdade e desigualdade. Dependendo do objetivo do estudo, podem ser definidos diferentes problemas de otimização para um mesmo sistema. Algumas características da formulação do FPO são mostradas a seguir (EL-HAWARY, 1996; CASTRONUOVO, 1997).

2.4.1. Variáveis do FPO

Se a tensão complexa em cada barra é conhecida, é possível calcular qualquer outra quantidade da rede. Por esta razão, a tensão complexa é considerada variável de otimização. Tapes de transformadores LTC e defasadores, capacitores, reatores regulados, etc., devem ser também considerados como variáveis de otimização do sistema. A este conjunto mínimo de variáveis de otimização é possível adicionar outras variáveis, dependendo do objetivo de estudo e do método de otimização utilizado. Algumas grandezas que podem ser adicionadas como variáveis de otimização são: potências ativas e reativas de geração, limite máximo de transmissão numa linha, transações contratadas entre barras, etc.

2.4.2. Restrições de Igualdade

O número mínimo de restrições de igualdade do FPO é determinado pelas equações da rede, tal como no problema de Fluxo de Potência. Adicionalmente, podem ser incluídas no problema de otimização do FPO restrições de igualdade que modelam características particulares de operação da rede (valores fixos de algumas variáveis ou combinação de variáveis do sistema).

2.4.3. Restrições de Desigualdade

As restrições de desigualdade são as limitações impostas a uma variável, ou conjunto de variáveis, do sistema. Em relação à sua função, elas podem ser classificadas em três grandes grupos:

Restrições físicas: São incluídas neste grupo as restrições impostas pelas limitações de capacidade dos componentes do sistema. Exemplos destas limitações podem ser: limites máximo e mínimo de geração de potência das unidades geradoras, limites nos valores dos taps, limites de transmissão de potência ativa nas linhas, etc.

Restrições operacionais: A prática de operação do sistema impõe limites que devem ser considerados na modelagem. Alguns exemplos destas restrições são: limites máximo e mínimo da magnitude da tensão nas barras, defasamento angular máximo entre barras, limites originados pela taxa de tomada de carga das unidades geradoras, etc.

Restrições de segurança: As restrições de segurança representam um grupo de restrições relacionadas a um conjunto de contingências determinadas pela análise de segurança em tempo real. A modelagem via análise de segurança fornece resultados para a operação num estado seguro do sistema(VARGAS *et al.*, 1993, SHEBLE, 1990).

Em relação à sua representação matemática no problema de otimização, as restrições podem ser divididas em duas classes:

Restrições simples: Este tipo de restrições consiste de limites nas variáveis de otimização, representadas na Equação (2.4).

$$x \leq x^M \quad (2.4)$$

Restrições funcionais: Este tipo de restrição é modelado como uma função das variáveis de otimização, na forma expressa pela Equação (2.5). Em geral, este tipo de restrição impõe condições mais severas à convergência dos métodos de otimização que as restrições simples.

$$h(x) \leq h^M \quad (2.5)$$

O grande número de restrições envolvidas, a impossibilidade de se identificar no início do processo de otimização as restrições ativas no ponto ótimo e a existência de numerosas restrições funcionais impõem condições críticas aos métodos de otimização a serem aplicados ao FPO.

2.4.4. Função Objetivo

A função objetivo representa a variável ou conjunto de variáveis que se deseja otimizar. Existe uma gama muito grande de funções objetivo passíveis de utilização na operação da rede elétrica, relacionadas ao objetivo do estudo. No FPO, algumas das mais utilizadas são (SOMAN *et al.*, 1994, PARKER, 1995, IRISARRI, 1996, BARBOZA, 1997, CASTRONUOVO, 1997):

Mínimo Custo de Potência Ativa: visa refletir a operação econômica da rede. O custo de produção da energia é expresso em função da potência ativa gerada pelas unidades. Esta função é não-linear em relação às tensões nas barras, sendo geralmente representada por uma função objetivo linear ou quadrática das potências ativas (e/ou reativas) de geração. Atualmente, estuda-se o efeito da representação dos custos de geração da potência reativa no sistema;

Minimização das Perdas de Potência Ativa: visa diminuir o valor total das perdas no sistema. Esta função é uma expressão não-linear das tensões nas barras e, similarmente à anterior, pode ser expressa em função de variáveis auxiliares;

Mínimo Desvio de uma Distribuição de Potências Ativas Pré-especificada: visa aproximar à geração do sistema a valores predeterminados. Esta função pode ser utilizada em despacho corretivo e na análise da operação considerando-se transações. Geralmente, esta função é representada através de uma equação quadrática das potências ativas de geração;

Mínimo Desvio de um Perfil de Tensões Pré-especificado: procura operar o sistema próximo a um ponto de operação predeterminado. Esta função é utilizada correntemente em despacho corretivo e é representada por uma equação quadrática dos módulos das tensões (variáveis do sistema);

Mínimo Corte de Carga: visa diminuir o corte de carga em situações de emergência e/ou estabelecer limites no carregamento do sistema. Esta função pode ser expressa através de uma função linear do parâmetro de incremento de carga;

Mínima Ação de Controle: procura estabelecer o mínimo número de controles necessários para uma operação próxima à ótima. Existem diversas formulações analíticas alternativas para modelar esta função objetivo.

As funções objetivo anteriores podem ser combinadas num único problema de otimização, através de uma abordagem multi-objetivo (CARVALHO, 1999, CARVALHO e SALGADO, 1997, YOKOYAMA *et al.*, 1988).

Uma vez modelado o FPO, deve-se encontrar uma metodologia que permita a sua resolução. Alguns conceitos são requeridos para a solução de problemas de otimização restritos. Nas seções seguintes, são apresentados os conceitos de função Lagrangeana, Dualidade, funções de Penalidade e as condições de otimalidade de primeira ordem.

2.5. A Função Lagrangeana

A utilização da função Lagrangeana é um artifício utilizado para transformar um problema de otimização Linear ou Não-Linear sujeito apenas a restrições de igualdade num problema de otimização irrestrita. Este artifício consiste em adicionar as restrições de igualdade à função objetivo, formando uma nova função objetivo, a qual é denominada de função Lagrangeana ou Lagrangeano do problema.

Suponhamos um problema de otimização com apenas restrições de igualdade, isto é

$$\text{Min. } f(\mathbf{x}) \quad (2.6)$$

$$\text{s.a. } g(\mathbf{x}) = 0 \quad (2.7)$$

A função Lagrangeana deste problema é definida pela equação (2.8).

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}' \mathbf{g}(\mathbf{x}) \quad (2.8)$$

onde: $L(\mathbf{x}, \boldsymbol{\lambda})$ é a função Lagrangeana; e $\boldsymbol{\lambda}'$ é o vetor transposto dos multiplicadores de Lagrange (ZANGWILL, 1969, CASTRONUOVO, 1997).

Como se observa na equação (2.8), as restrições de igualdade são adicionadas à função objetivo pré-multiplicadas por novas variáveis, os chamados *multiplicadores de Lagrange*. No problema de otimização modificado são variáveis: as originais (ou primais) \mathbf{x} e as adicionais $\boldsymbol{\lambda}$ (denominadas de duais).

A seguir é mostrado um teorema que formula as condições para a existência dos multiplicadores de Lagrange.

2.5.1. Teorema da Existência dos Multiplicadores de Lagrange

Sejam

$$f(x_i) \text{ e } g_j(x_i), i= 1, \dots, n, j=1, \dots, m, m < n$$

funções definidas em \mathbb{R}^n com derivadas parciais de primeira ordem todas contínuas em um conjunto aberto \mathbf{S} , tal que:

$$\mathbf{S} = \{ \mathbf{x}_i / g_j(x_i) = 0, i=1 \dots n, j=1 \dots m \}$$

Se existe um ponto ótimo \mathbf{x}^0 no qual $f(\mathbf{x}^0) \leq f(\mathbf{x})$ para todo $\mathbf{x} \in \mathbf{S}$ e se o posto do Jacobiano

$$\begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \dots & \frac{\partial g_m}{\partial x_1} \\ \dots & \dots & \dots \\ \frac{\partial g_1}{\partial x_n} & \dots & \frac{\partial g_m}{\partial x_n} \end{bmatrix} \quad (2.9)$$

no ponto x^o é m , então existem m números reais λ_i , não todos nulos, tais que

$$\nabla f(x^o) + \lambda' * \nabla g(x^o) = 0 \quad (2.10)$$

A prova deste teorema pode ser encontrada no Apêndice 11 da ref. (AOKI, 1971).

A utilização da função Lagrangeana está associada ao conceito de Dualidade.

2.6. Dualidade

Na resolução do problema (2.8) existem dois tipos de variáveis, das quais deve-se conhecer o valor no ponto ótimo. Suponha-se que existe na função Lagrangeana um ponto de sela (ZANGWILL, 1969), como o representado na Figura 2, tal que;

$$L(x, \lambda^o) \leq L(x^o, \lambda^o) \leq L(x^o, \lambda) \quad \text{para } x \in X \quad \text{e } \lambda \in \Lambda \quad (2.11)$$

onde a região $X \subset R^n$, com n o número de variáveis; e $\Lambda = \{ \lambda = (\lambda_1, \dots, \lambda_m)^t, i=1, \dots, m \}$, com m multiplicadores de Lagrange, é o conjunto dos multiplicadores de Lagrange.

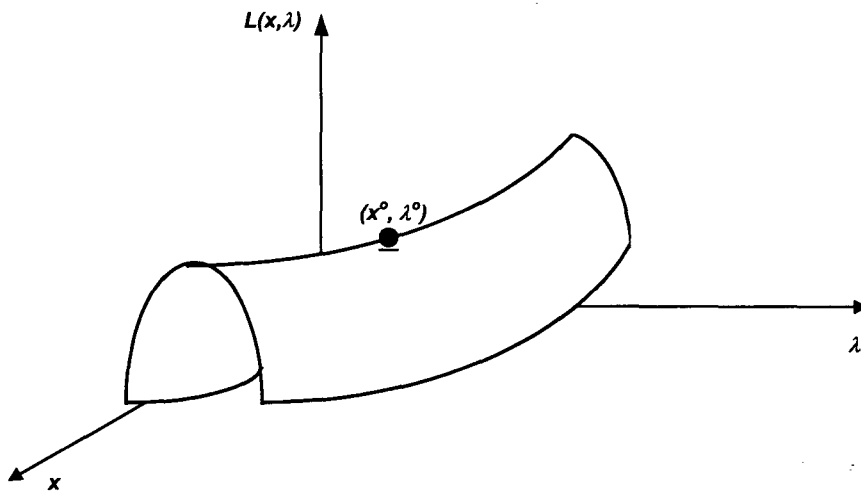


Figura 2: Representação do Ponto de Sela.

Então, define-se a *função primal* como

$$L_*(x) = \max_{\lambda \in \Lambda} L(x, \lambda) \quad (2.12)$$

e a *função dual* como

$$L^*(\lambda) = \min_{x \in X} L(x, \lambda) \quad (2.13)$$

Na função primal λ é considerado um parâmetro fixo, que deve ser pre-determinado. As variáveis primais x (únicas presentes na função primal) são calculadas através de um problema de otimização. O *problema primal* consiste então na determinação do ótimo x^* , para o qual:

$$L_*(x^0) = \min_{x \in X} L_*(x) \quad (2.14)$$

De forma similar, o *problema dual* diz respeito à obtenção do ótimo λ^* tal que;

$$L^*(\lambda^0) = \max_{\lambda \in \Lambda} L^*(\lambda) \quad (2.15)$$

Substituindo a expressão (2.12) na (2.14), resulta:

$$\min_{\mathbf{x} \in \mathbf{X}} L(\mathbf{x}) = \min_{\mathbf{x} \in \mathbf{X}} \max_{\lambda \in \Lambda} L(\mathbf{x}, \lambda) = \min_{\mathbf{x} \in \mathbf{X}} f(\mathbf{x}) \quad (2.16)$$

$\mathbf{x} \in \mathbf{X}$ sujeito a $\mathbf{g}(\mathbf{x}) = 0$

A equação (2.16) expressa que o problema de otimização original (terceiro membro da equação), através da função Lagrangeana, pode ser representado pela resolução simultânea de dois problemas irrestritos, um de minimização e outro de maximização (segundo membro da equação).

Um resultado similar pode ser obtido da análise da função e problema duais. Pode-se afirmar, na existência de um ponto de sela na Função Lagrangeana (ZANGWILL, 1969), que:

$$\min_{\mathbf{x} \in \mathbf{X}} \max_{\lambda \in \Lambda} L(\mathbf{x}, \lambda) = \max_{\lambda \in \Lambda} \min_{\mathbf{x} \in \mathbf{X}} L(\mathbf{x}, \lambda) \quad (2.17)$$

O teorema apresentado a seguir garante as condições de convergência dos problemas primal e dual para programação convexa.

2.6.1. Teorema da Existência das Soluções Primal e Dual

Dado um conjunto aberto $\mathbf{X} \subset \mathbf{R}^n$, e uma solução $\mathbf{x}^* \in \mathbf{X}$ do problema

$$\text{Min. } f(x) \quad (2.18)$$

$$\text{s.a. } g(x) = 0 \quad (2.19)$$

Se $f(x)$ é convexa e $g(x)$ são côncavas, se o problema associado à (2.18) e (2.19) é expresso como

$$\text{Minimizar } L(x, \lambda) = f(x) + \lambda' * g(x) \quad (2.20)$$

e se x^o satisfaz as condições de Karush-Kuhn-Tucker, então

1- x^o é a solução ótima do Problema de Programação Não-Linear (2.18)-(2.19)

2- x^o é a solução ótima do Problema de Programação Não-Linear (2.20)

A demonstração deste teorema está na pág. 49 do livro (ZANGWILL, 1969). Deve-se observar que nesta referência resolve-se um problema de maximização, o qual no presente texto está expresso como de minimização.

2.6.2. Classificação dos Métodos de Otimização para Problemas Restritos.

Da análise anterior, os algoritmos de resolução de problemas de otimização restritos podem ser classificados em três categorias, de acordo com o problema a ser resolvido:

Algoritmos Primais: resolvem um problema do tipo expresso na equação (2.14). As únicas variáveis consideradas no problema de otimização são as primais.

Algoritmos Duais: resolvem o problema dual (2.15). As variáveis consideradas nesta classe de problemas são os multiplicadores de Lagrange.

Algoritmos Primais-duais: visam resolver simultaneamente os problemas Primal e Dual.

2.7. As Funções de Penalidade.

O uso de funções de Penalidade e de Barreira constitui uma abordagem historicamente muito utilizada (FIACCO e McCORMICK, 1968), principalmente nos métodos de minimização seqüencial. Existem distintas versões para seu uso; uma revisão daquelas mais utilizadas é apresentada no Capítulo 12 da referência (FLETCHER, 1987). Suponha-se que o problema considerado possui restrições de igualdade (equações (2.37) e (2.38)) e restrições de desigualdade. Então ele pode ser representado através das equações (2.21) e (2.22).

$$\text{Min. } L(x, \lambda) = f(x) + \lambda' * g(x) \quad (2.21)$$

$$\text{s.a } h(x) \leq 0 \quad (2.22)$$

As restrições de desigualdade ativas podem ser incorporadas à função objetivo através de funções de penalidade, resultando:

$$L_2(x, \lambda) = f(x) + \lambda' * g(x) + \frac{1}{2} * C_p * h(x)' * h(x) \quad (2.23)$$

onde C_p é o coeficiente da penalidade.

A penalidade é formada da soma dos quadrados das violações nas restrições de desigualdade afetada pelo coeficiente de penalidade. Este parâmetro determina o aumento na penalidade (ZANGWILL, 1969).

Suponha-se que na solução ótima $(x^o, \lambda^o, \sigma^o)$ existem restrições de desigualdade $h(x^o)$ satisfeitas na igualdade. Por conseguinte, pode-se dividir o conjunto de restrições de desigualdade $h(x)$ nos subconjuntos $h_1(x)$ (ativo na solução) e $h_2(x)$ (restrições que continuam sendo de desigualdade na solução). No ponto ótimo, às restrições ativas pode ser aplicado o Teorema de Existência dos Multiplicadores de Lagrange como enunciado anteriormente, devendo ser satisfeita a equação (2.24).

$$\frac{1}{2} * C_{p_i} * h_i(x^o) \rightarrow \lambda_i^o \quad (2.24)$$

onde Cp_1 é o subconjunto dos coeficientes de penalidade das restrições $h_1(x)$; e λ_1^o é o vetor dos novos multiplicadores de Lagrange associados às restrições de desigualdade satisfeitas na igualdade $h_1(x)$.

Sendo $h_1(x^o) = 0$, pode-se supor que $\lambda_1^o \neq 0$. A fim de satisfazer a equação (2.24), na solução os coeficientes de penalidade $Cp_1 \rightarrow \infty$. A utilização de coeficientes de penalidade muito grandes piora o condicionamento das matrizes envolvidas no processo de solução, provocando problemas para a convergência (EL-BAKRY *et al.*, 1996).

Da análise anterior, observa-se que os parâmetros Cp_1 devem satisfazer dois critérios no processo de solução:

- Na convergência global, os coeficientes Cp_1 penalizam as violações nas restrições de desigualdade;
- Na convergência local (próxima ao ponto solução), os coeficientes Cp_1 devem ser determinados para aproximar $\frac{1}{2} * Cp_1 * h_1(x^o)$ ao multiplicador de Lagrange λ_1^o .

Estas funções podem ser diferenciadas considerando-se as restrições ativas na solução $h_1(x)$ como igualdades, e penalizar exclusivamente aquelas restrições não ativas na solução. Esta consideração resulta numa formulação da denominada função Lagrangeana aumentada (HESTENES, 1969, EL-BAKRY, 1996), reduzindo o problema de condicionamento das matrizes. No problema considerado, o Lagrangeano resulta na equação (2.25).

$$L_3(x, \lambda, \lambda_1) = f(x) + \lambda' * g(x) + \lambda_1' * h_1(x) + \frac{1}{2} * Cp * h_2(x)' * h_2(x) \quad (2.25)$$

2.8. As Condições de Otimalidade de Primeira Ordem.

As equações de otimalidade de primeira ordem são também conhecidas como condições de Karush-Kuhn-Tucker (KKT) (KUHN e TUCKER, 1951, LUENBERGER, 1965). Estas condições devem ser satisfeitas em qualquer ponto ótimo restrito, local ou global, dos problemas de programação linear e da maioria dos problemas de programação não-

linear. As condições de KKT formam a base para o desenvolvimento de muitos algoritmos computacionais e são utilizadas como critérios de convergência de variados métodos. Estas equações dão as condições necessárias para que um ponto seja considerado candidato a ótimo ou mínimo local (FLETCHER, 1987).

Para o problema não-linear de otimização com restrições de igualdade e desigualdade (equações (2.1) à (2.3)), o Lagrangeano pode ser representado como:

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda' * \mathbf{g}(\mathbf{x}) + \pi' * \mathbf{h}(\mathbf{x}) \quad (2.26)$$

onde π é o vetor dos multiplicadores de Lagrange das restrições de desigualdade.

As condições de otimalidade de KKT podem ser expressas como:

$$\nabla f(\mathbf{x}) + \nabla \mathbf{g}(\mathbf{x}) * \lambda + \nabla \mathbf{h}(\mathbf{x}) * \pi = \mathbf{0} \quad (2.27)$$

$$\varphi - \pi = \mathbf{0} \quad (2.28)$$

$$\mathbf{g}(\mathbf{x}) = \mathbf{0} \quad (2.29)$$

$$\mathbf{h}(\mathbf{x}) \leq \mathbf{0} \quad (2.30)$$

$$[\varphi] * \mathbf{h}(\mathbf{x}) = \mathbf{0} \quad (2.31)$$

$$\pi \geq \mathbf{0} \quad (2.32)$$

onde: φ é o vetor dos multiplicadores de complementaridade das equações de desigualdade; e $[..]$ representa a matriz diagonal dos elementos considerados.

A equação (2.27) garante um gradiente do Lagrangeano (equação (2.26)) nulo no ponto candidato ao ótimo. A equação (2.28) diz que, neste ponto, os multiplicadores de Lagrange e os multiplicadores de complementaridade das restrições de desigualdade são iguais. As equações (2.29) e (2.30) tornam factível a solução. Finalmente, a equação de complementaridade estrita (2.31) permite considerar simultaneamente todas as restrições de desigualdade, sejam elas ativas ou não na solução, ao custo de incrementar o número de variáveis duais. Nesta última equação, dois casos podem ser analisados:

- Para aquelas restrições de desigualdade satisfeitas na igualdade no ponto ótimo, podem-se considerar multiplicadores $\varphi \neq 0$;
- Nas restrições não satisfeitas no limite no ponto candidato ao ótimo ($h(x) < 0$), os multiplicadores de complementaridade são nulos.

É possível formular condições necessárias de otimalidade de segunda ordem. Entretanto, ao serem em geral de difícil avaliação e utilização em sistemas de potência, não são consideradas no presente trabalho. Estas condições podem ser estudadas com detalhes nos trabalhos de FLETCHER (1987), BERTSEKAS (1996) e outros.

2.9. Métodos de Solução do FPO.

O FPO (equações (2.1) à (2.3)) é um problema de otimização estático, não-linear, de grande porte, ao qual podem ser aplicados diferentes métodos de resolução. Algumas das metodologias utilizadas para sua resolução são mostradas a seguir.

2.9.1. Metodologias Baseadas em Programação Linear

A Programação Linear é o ramo da Otimização Matemática que estuda os métodos de solução de problemas com função objetivo e restrições lineares em relação às variáveis de otimização. O FPO é um problema não-linear, que pode ser aproximado através de linearizações sucessivas. As equações (2.1) à (2.3), originais do problema, são resolvidas como uma sucessão de aproximações lineares, da forma:

$$\text{Min. } f'(x^o + \Delta x) \quad (2.33)$$

$$\text{s.a } g'(x^o + \Delta x) = 0 \quad (2.34)$$

$$h'(x^o + \Delta x) \leq 0 \quad (2.35)$$

onde: x^o é o valor inicial de x ; Δx é a variação em relação ao ponto inicial; e f' , g' e h' são aproximações lineares das funções não-lineares originais.

Cada linearização calcula a direção ao ponto ótimo Δx através da linearização da função objetivo e das restrições. Entretanto, a solução iterativa do problema linear (equações

(2.33) à (2.35)) não garante a solução do problema não-linear original (equações (2.1) à (2.3)). Portanto, deve-se executar um Fluxo de Potência entre cada linearização. Os passos básicos para a resolução de um FPO via Programação Linear são resumidos no Algoritmo 2.1 (STOTT e MARINHO, 1979).

Algoritmo 2.1. FPO via Programação Linear

1. Resolução do Fluxo de Potência Não-Linear.
 2. Linearização da função objetivo e restrições de desigualdade.
 3. Linearização do modelo de rede incremental.
 4. Resolução de um problema de Otimização Linear.
 5. Resolução do Fluxo de Potência Não-Linear.
 6. Teste de Convergência. SIM: Fim. NÃO: Retornar ao Passo 2.
-

As principais vantagens da utilização de metodologias baseadas na Programação Linear no FPO são (RISTANOVIC, 1996):

- Eficiente detecção de casos sem solução real;
- Facilidade na resolução de problemas de FPO com análise de segurança;
- Tempos relativamente reduzidos de resolução;
- Facilidade na análise matemática das condições de convergência dos subproblemas.

Alguns dos métodos mais comuns de solução do FPO via Programação Linear são:

2.9.1.1. O Método Simplex

O método Simplex procura, desde uma solução possível na periferia da região de convergência, outra solução que possua um valor de função objetivo inferior, também com restrições de desigualdade ativas. A evolução do método modificou o algoritmo Simplex original, com variantes que consideram as variáveis duais (Dual Simplex) e/ou

procuram reduzir o número de operações requeridas pelo método (Dual Simplex Revisado). A bibliografia sobre os métodos Simplex é muito extensa, podendo-se destacar os trabalhos de LUENBERGER (1965), FLETCHER (1987) e DANTZIG (1963).

Existe um razoável número de aplicações destes algoritmos no FPO, devido à sua robustez e simplicidade. A maioria deles visa resolver o despacho de potência ativa na rede (HUNEAULT, GALIANA, 1991). O primeiro algoritmo para resolução de problemas reais de grande porte baseado em programação linear foi desenvolvido pela EDF, o qual considerava a maioria das tarefas da operação da rede (MERLIN, 1972). Exemplos de aplicação do método Simplex ao FPO são os trabalhos de STOTT e MARINHO (1979), STOTT e HOBSON (1977a e 1977b), NABONA e FERRIS (1973) e NICHOLSON e STERLING (1973).

2.9.1.2. Métodos Baseados no Vetor Gradiente.

Os algoritmos que utilizam o vetor gradiente para a determinação da direção de busca podem ser aplicados tanto em programação linear como não-linear. Devido à possibilidade de uma representação mais geral, eles são analisados posteriormente na análise dos algoritmos não-lineares.

2.9.1.3. Métodos de Pontos Interiores para Programação Linear

Os métodos de Pontos Interiores surgem nos meados da década dos anos oitenta, com o trabalho de KAMARKAR (1984). O algoritmo proposto nesta referência era primal, de convergência polinomial e propõe um caminho central ao ponto ótimo, interior à região viável formada pelas restrições de desigualdade (FLETCHER, 1987). Desde a sua formulação, os métodos de Pontos Interiores constituem uma das mais intensivas áreas de pesquisa na Otimização Matemática, tanto em Programação Linear como Não-Linear. O algoritmo original foi modificado a fim de considerar as variáveis duais, resultando numa formulação primal-dual. Existem três famílias principais de métodos de Pontos Interiores (WRIGHT, 1997):

Métodos de Redução Potencial:

Continuação das pesquisas de KAMARKAR (1984), estes métodos visam minimizar em cada passo do processo iterativo uma função potencial (ou função de mérito) predefinida. Apesar de não oferecerem características teóricas de rápida convergência, intensivos testes computacionais demonstraram que este tipo de método é geralmente mais eficiente na prática do que aqueles baseados na metodologia Simplex (YU, NESTEROV, 1996). A

diferença entre as características teóricas e práticas do método podem ser explicadas pela utilização de uma estratégia apropriada de busca linear, a qual pode decrescer o valor do potencial numa quantidade maior à garantida pela análise de pior caso. Algumas novas propostas nesta área são as de FORSGREN e GILL (1996), YU e NESTEROV (1996) e WANG *et al.* (1995). Para aplicações deste método ao FPO, pode-se consultar (CARVALHO, 1999).

Métodos "path-following".

O seguinte avanço teórico na área dos métodos de Pontos Interiores é devido a RENEGAR (1988), que prova a convergência de um algoritmo polinomial baseado no método de Newton, constituindo um exemplo de um método "path-following". Os algoritmos "path-following" visam atingir o ponto ótimo através de uma trajetória próxima ao caminho central (WRIGHT, 1997). A partir das boas características teóricas de convergência, têm-se implementado distintas variantes destes métodos, segundo o critério de vizinhança à trajetória central utilizado. As metodologias variam desde critérios simples de seleção da trajetória (EL-BAKRY, 1996), até algoritmos de Máximo Passo (HERSKOVITS, 1998, GONZAGA e BONNANS, 1996), passando por aqueles do tipo Preditor-Corretor (MEHROTRA, 1992). A maioria destes métodos têm sido estudados em problemas de Programação Linear, pela simplicidade na análise das condições de convergência. No presente trabalho, realiza-se um estudo de alguns destes métodos aplicados ao problema não-linear de FPO. Uma análise mais detalhada destes métodos pode ser encontrada no Capítulo 3.

Métodos de Ponto Inicial Não-Factível.

Os algoritmos lineares de Pontos Interiores requerem um ponto inicial interior à região factível. Como este ponto inicial nem sempre é fácil de se encontrar, numerosos pesquisadores têm desenvolvido algoritmos que não requerem esta condição. Uma interessante explicação destes métodos pode ser encontrada no trabalho de WRIGHT (1997). Como em Programação Linear o ponto ótimo deve se encontrar na periferia da região viável, pesquisadores têm proposto uma trajetória externa à região de convergência, em métodos denominados como de Pontos Exteriores. Trabalhos nessa área são os de ANDRUS e SCHAFERKOTTER (1996) e ANDRUS (1992).

No Apêndice B da referência (WRIGHT, 1997) apresenta-se uma lista dos software de Pontos Interiores disponíveis na Internet. Nessa listagem, são fornecidas algumas informações do algoritmo computacional, método utilizado e o endereço WEB.

As equações representativas do FPO são não-lineares e em alguns casos é difícil a sua aproximação por funções lineares. Por conseguinte, muitos pesquisadores têm optado por resolver diretamente o problema não-linear, através de técnicas de Programação Não-Linear.

2.9.2. Metodologias Baseadas em Programação Não-Linear

A Programação Não-Linear representa mais precisamente o modelo físico do FPO, ao custo de ser mais onerosa em termos computacionais. Nesta classe de métodos, a função objetivo e/ou as restrições são representadas por equações não-lineares. Alguns dos métodos de Programação Não-Linear utilizados na resolução de problemas de FPO são listados a seguir:

2.9.2.1. Programação Quadrática Seqüencial.

Apesar de ser classificada como pertencente à Programação Não-Linear, esta classe de algoritmos é intermediária entre aqueles de Programação Linear e os de estrita programação Não-Linear. A Programação Quadrática Seqüencial, semelhante à Programação Linear Seqüencial, resolve o problema não-linear original através de aproximações. Entretanto, nestas aproximações o sistema físico é modelado através de uma função objetivo de segundo grau e restrições lineares (ZANGWILL, 1969, FLETCHER, 1987). No caso do FPO, o algoritmo utilizado é similar ao da Programação Linear Sucessiva (Algoritmo 2.1), utilizando-se no Passo 2 uma função objetivo quadrática. Alguns trabalhos na área são os de BURCHETT, HAPP e VIERATH (1984), MOMOH (1996).

2.9.2.2. Baseados no Vetor Gradiente.

Estes foram os primeiros e mais conhecidos algoritmos de ascendência. Os algoritmos de ascendência (método do gradiente, Newton, Pontos Interiores, etc.) possuem uma estrutura comum: partindo de um ponto inicial, determina-se, através de uma certa regra, uma direção de busca, e após isto, move-se nesta direção até satisfazer um determinado critério (Figura 3).

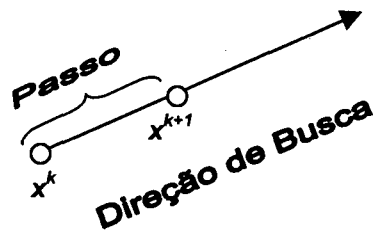


Figura 3: Algoritmos de Ascendência.

Nos algoritmos baseados no vetor gradiente, a direção de busca é definida com auxílio do vetor de primeiras derivadas. Um fator de passo é calculado, o qual determina a magnitude do incremento nas variáveis do processo. O método do gradiente é definido pelo algoritmo iterativo (FRITZSCHE, 1978):

$$\mathbf{x}^{k+1} = \mathbf{x}^k + a_p * \nabla f(\mathbf{x}^k) \quad (2.36)$$

onde: $\nabla f(\mathbf{x}^k)$ é o vetor gradiente de $f(\mathbf{x})$, calculado no ponto \mathbf{x}^k ; a_p é o fator de passo; e \mathbf{x}^k e \mathbf{x}^{k+1} é o valor das variáveis nas iterações k e $(k+1)$, respectivamente.

Numerosas versões deste método têm sido desenvolvidas, dentre elas (ZANGWILL, 1969) os algoritmos de *Gradiente Reduzido*, *Projetado* e *Conjugado*.

A principal vantagem da utilização de métodos de gradiente no FPO é a sua simplicidade, já que as metodologias baseadas nessas técnicas podem ser utilizadas como uma extensão de métodos de Fluxo de Potência convencionais. O problema resultante é reduzido e relativamente fácil de ser resolvido. Segundo HUNEULT e GALIANA (1991), duas variantes dominam a literatura: o método de gradiente Reduzido Generalizado (CARPENTIER, 1972) e o método proposto por DOMMEL e TINNEY (1968). Este último resolve as equações de Karush-Kuhn-Tucker usando uma combinação do método de gradiente e funções de penalidade para as restrições de desigualdade violadas. A utilização destas funções de penalidade diminui em parte a eficiência da formulação de Dommel e Tinney, sendo entretanto um dos métodos mais utilizados no desenvolvimento do FPO.

2.9.2.3. O Método de Newton.

Esta classe de métodos de ascendência, originalmente formulada para a resolução de problemas de otimização irrestrita, pode ser aplicado em Programação Não-Linear com apenas restrições de igualdade, como expresso nas equações (2.37) e (2.38).

$$\text{Min. } f(\mathbf{x}) \tag{2.37}$$

$$\text{s.a. } \mathbf{g}(\mathbf{x}) = \mathbf{0} \tag{2.38}$$

O método de Newton é formulado, adicionando-se inicialmente as restrições de igualdade à função objetivo através de multiplicadores de Lagrange, formado a *função Lagrangeana* (BERTSEKAS, 1996):

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}' * \mathbf{g}(\mathbf{x}) \tag{2.39}$$

O ponto candidato a ótimo da função Lagrangeana (2.39) é obtido num processo iterativo da forma:

$$(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) = (\mathbf{x}^k, \boldsymbol{\lambda}^k) - a_p * (\nabla^2 L(\mathbf{x}^k, \boldsymbol{\lambda}^k))^{-1} * \nabla L(\mathbf{x}^k, \boldsymbol{\lambda}^k) \tag{2.40}$$

onde: $\nabla^2 L(\mathbf{x}^k)$ e $\nabla L(\mathbf{x}^k)$ são as matrizes de segundas derivadas (Hessiana) e o vetor de primeiras derivadas (gradiente do Lagrangeano), avaliados no ponto \mathbf{x}^k .

A forma pura do método de Newton considera fatores de passo a_p unitários. Entretanto, em algumas versões este fator pode ser recalculado através de uma minimização unidimensional na direção de busca ou definida pela regra de Armijo (BERTSEKAS, 1996).

A equação (2.40) é resolvida assumindo que $(\nabla^2 L(\mathbf{x}^k))^{-1}$ existe e que a direção

$$\mathbf{d}^k = -(\nabla^2 L(\mathbf{x}^k))^{-1} \nabla L(\mathbf{x}^k) \tag{2.41}$$

é uma direção descendente ($\mathbf{d}^k * \nabla L(\mathbf{x}^k) < 0$). Na prática, a direção é obtida através da resolução de um sistema linear da forma:

$$\nabla^2 L(\mathbf{x}^k) * \mathbf{d}^k = -\nabla L(\mathbf{x}^k) \quad (2.42)$$

O critério de convergência para o algoritmo de Newton é baseado nas condições de otimalidade de primeira ordem de Karush-Kuhn-Tucker (KKT) (SANCHES,1983, AOKI, 1971). Como visto anteriormente, estas condições estabelecem as equações que devem ser satisfeitas num ponto candidato a solução ótima \mathbf{x}^* .

O algoritmo de resolução via método de Newton (considerando-se somente as restrições de igualdade) pode ser resumido em:

Algoritmo 2.2. Forma Pura do Método de Newton

1. Inicialização das variáveis primais e duais (\mathbf{x}, λ) .
 2. Avaliação do gradiente do Lagrangeano no ponto atual $(\nabla L(\mathbf{x}^k))$.
 3. Teste de Convergência através das condições de KKT: Convergência: Fim. Não Convergência: Ir ao Passo 4.
 4. Obtenção da direção de busca, mediante a resolução do sistema linear (2.42).
 5. Atualização da solução corrente.
 6. Retornar ao Passo 2
-

Além das restrições de igualdade, geralmente os problemas de otimização possuem restrições de desigualdade. A consideração destas restrições no método de Newton pode ser feita utilizando-se funções de Penalidade.

Na modelagem de problemas não-lineares, não são conhecidas "a priori" as restrições ativas no ponto ótimo, podendo ser utilizadas técnicas de Sistemas Especialistas e/ou minimizações sequenciais para esta finalidade. A primeira destas técnicas visa determinar, a partir do conhecimento prévio do sistema, quais restrições serão ativas no

ponto ótimo. As minimizações seqüenciais visam adicionar em cada iteração aquelas restrições violadas e retirar aquelas que deixam de ser ativas na iteração, até atingir o ponto ótimo. Esta última técnica possui as seguintes desvantagens (FLETCHER, 1987, EL-BAKRY *et al.*, 1996, CASTRONUOVO, 1997):

- Modificação na dimensão do sistema linear a ser resolvido em cada iteração;
- Determinação do parâmetro de penalidade a ser utilizado;
- Possível deterioração do número de condicionamento da matriz de coeficientes do sistema linear a ser resolvido em cada iteração, principalmente na vizinhança do ponto ótimo.

No caso do FPO, a proposição da utilização do método de Newton por SUN *et al.* (1984) foi marcante na área de otimização aplicada, formando a base para muitos métodos atualmente utilizados (HUNEAULT e GALIANA, 1991, MONTICELLI e LIU, 1992). Tipicamente, estas metodologias resolvem as equações de KKT usando o método de Newton para as restrições de igualdade, com a adição de controles para as restrições de desigualdade. Entretanto, a manipulação das restrições de desigualdade se manteve como uma dificuldade a ser superada.

Uma variante do método de Newton considera aproximações da matriz de segundas derivadas $\nabla^2 L(x^k)$, formando os métodos Quasi-Newton (BERTSEKAS, 1996, FRITZSCHE, 1978, FLETCHER, 1987). Estes métodos são utilizados em problemas de pequenas dimensões ou quando a formulação do problema de otimização gera uma matriz Hessiana densa. Nestes casos, a determinação dos valores da matriz pode demandar elevado esforço computacional, incentivando a utilização de cálculos aproximados. Os métodos Quasi-Newton requerem, em geral, o uso de pré-condicionadores para favorecer a convergência. Algumas referências para este grupo de métodos são os trabalhos de FLUECK e CHIANG, 1998, GALIANA *et al.* (1994), MORI *et al.* (1996), COVA *et al.* (1987), GIRAS e TALUKDAR (1981) e TALUKDAR e GIRAS (1982).

2.9.2.4. Algoritmos de Pontos Interiores Para Programação Não-Linear.

Esta classe de algoritmos propõe a modificação das condições de otimalidade de KKT durante o processo iterativo, na consideração das restrições de desigualdade. O problema resultante é um sistema de equações não lineares, resolvido via método de Newton. Neste caso, não se requer a determinação do conjunto das restrições de desigualdade ativas na solução.

As modificações introduzidas pelo método de Pontos Interiores na aplicação das condições de otimalidade de KKT são:

- Transformação de todas as restrições de desigualdade em restrições de igualdade, através do uso de variáveis de folga;
- Relaxação no processo iterativo da equação de complementaridade estrita, por um parâmetro $\mu \geq 0$.

As novas equações de otimalidade, conhecidas como *condições de KKT perturbadas*, são resolvidas pelo método de Newton. O ponto candidato ao ótimo é determinado via resolução de um problema de otimização com apenas restrições de igualdade.

O algoritmo básico de resolução de um problema via Método de Pontos Interiores pode ser resumido no Algoritmo 2.3.

Algoritmo 2.3: Método de Pontos Interiores - Algoritmo Básico

1. Inicialização das variáveis primais e duais.
 2. Teste de Convergência através das condições de KKT: SIM: Fim. NÃO: Ir ao Passo 3.
 3. Escolha de um $\mu \geq 0$ e perturbação das condições de KKT
 4. Obtenção da direção de busca.
 5. Atualização da solução corrente.
 6. Retornar ao Passo 2
-

De forma similar aos algoritmos de Pontos Interiores para Programação Linear, os métodos para programação não-linear podem ser classificados em metodologias de Redução Potencial e "path-following". Os métodos de Pontos Interiores para PNL não requerem em geral um ponto factível para iniciar o processo de convergência, podendo partir de qualquer ponto inicial. Algoritmos mais detalhados, e as equações correspondentes, são objeto de estudo no Capítulo 3 deste trabalho.

Num passado recente, foi proposta a solução do problema do FPO através dos métodos de Pontos Interiores para PNL. Nestas abordagens, programas computacionais foram desenvolvidos, com potencial suficiente para competir com os métodos tradicionalmente usados para resolver problemas de grande porte. Algumas aplicações de Métodos de Pontos Interiores em FPO para sistemas de grande porte foram realizadas por GRANVILLE (1994), WU *et al.* (1994), BARBOZA (1997), CARVALHO (1999), IRRISARI *et al.* (1996), WEI *et al.* (1998) e WU, DEBS, e MARSTEN (1994).

2.10. Desafios do FPO.

Carpentier propôs a formulação geral, não-linear do Despacho Econômico incluindo limites nas tensões e outras restrições operacionais (CARPENTIER, 1962). A formulação original foi posteriormente estendida para outros problemas. Na atualidade, todo problema que envolve a determinação do estado estacionário de um sistema de potência elétrico, com otimização simultânea de um índice de desempenho, é considerado um problema de FPO. Isto é, as equações do sistema são resolvidas ajustando os controles disponíveis de modo a minimizar uma função objetivo sujeita a restrições operacionais e de segurança. Todas as classes de problemas de FPO são considerados subconjuntos deste problema geral(PAPALEXOPOULOS, 1996).

Uma formulação apropriada do problema e uma efetiva metodologia de solução são necessárias para a real aplicação do FPO em sistemas de potência. Nos dias de hoje, o FPO é implementado na maioria dos novos Centros de Operação de Sistemas. Entretanto, ele é efetivamente operacional em poucos sistemas (RAMANATHAN, 1996). Alguns dos desafios que o FPO têm que superar, a fim de se converter numa ferramenta confiável para a operação em tempo real, são sintetizados a seguir.

2.10.1. Requisitos de Resposta Temporal.

O ponto de operação de um sistema de potência varia continuamente no tempo, algumas vezes lentamente e em outras ocasiões rapidamente ou até abruptamente em condições de emergência. Tais mudanças são originadas pela evolução das cargas nas barras, modificação nas variáveis de controle, mudanças nas restrições de operação e possíveis alterações na topologia da rede. Tais mudanças requerem reduzidos tempos de cálculo para aplicações "on-line". Este requisito é especialmente importante quando repetidas soluções do FPO são necessárias para um ponto de operação, como acontece na modelagem das variáveis discretas do sistema de potência através de variáveis contínuas e/ou na consideração das transações econômicas.

2.10.2. Robustez da Solução.

O FPO deve produzir soluções consistentes para sua utilização nos processos de decisão pelos operadores de sistema de potência. Nesta área, dois requisitos são críticos para o FPO: a) Insensibilidade ao ponto inicial; e b) obtenção da solução para regiões de convergência estreitas ou com numerosas restrições.

Alguns autores têm analisado a robustez da solução do FPO. PAPALEXOPOULOS *et al.* (1989) analisam o efeito da inicialização, da formulação desacoplada e da discretização das variáveis para sistemas de grande porte. Nesse trabalho, é resolvido um FPO baseado em técnicas de segunda ordem, e se estuda a robustez da metodologia mediante a simulação de diferentes casos num sistema de 1500 barras. MONTICELLI e LIU (1992) estudam algumas causas de não-convergência no FPO via método de Newton, considerando as singularidades da matriz de primeiras derivadas. ALMEIDA e GALIANA (1996) propõem uma análise sistemática dos casos críticos no FPO. A metodologia utiliza um FPO não-linear parametrizado, com resultados em sistemas até 118 barras. Pelas características não-lineares do FPO, uma análise completa das causas de não-convergência resulta num problema de grande complexidade, difícil de ser resolvido em aplicações "on-line".

Como o FPO é um problema de grande porte com numerosos limites operacionais, a metodologia de solução deve ser rápida e eficiente no controle das restrições de desigualdade, o maior subconjunto de restrições presentes no sistema de potência.

2.10.3. Número de Controles Afetados na Solução.

A maioria dos programas de FPO utiliza todos os controles existentes no sistema a fim de obter a solução ótima. Entretanto, na prática é muitas vezes recomendável diminuir o número de controles a serem modificados para se atingir a solução, ainda que a solução obtida seja sub-ótima. O problema consiste em se determinar o melhor conjunto de controles a serem modificados, dentre os existentes na rede. TINNEY *et al.* (1988) propõem a ordenação dos controles pela sua efetividade na obtenção de uma solução próxima à ótima. GUAN *et al.* (1995) utilizam métodos "fuzzy" para determinar este conjunto de controles ótimos. Entretanto, este é um desafio que o FPO ainda enfrenta, devido à formação de um problema combinatorial.

2.10.4. Modelagem das Variáveis Discretas.

O FPO possui variáveis discretas na sua formulação, entre elas: os taps dos transformadores, bancos de compensação chaveáveis, e a conexão ou desconexão de elementos do sistema de transmissão. A consideração destas variáveis no seu caráter discreto converte o FPO num problema de otimização misto (contínuo-discreto), de grande complexidade, com tempos de solução ordens superiores às dos problemas contínuos (RAMANATHAN, 1996). Na atualidade, a maioria dos algoritmos de FPO tratam os controles discretos como variáveis contínuas durante o processo de resolução. Obtida a solução, as variáveis discretas são movidas à posição discreta mais próxima da ótima. Este procedimento fornece soluções sub-ótimas aceitáveis sob o ponto de vista do tempo de cálculo. Espera-se que o contínuo avanço das técnicas de otimização para variáveis discretas forneça soluções ótimas em tempos mais reduzidos que os atuais.

2.10.5. Inclusão das Transações Econômicas Bilaterais no FPO.

Nos últimos tempos, a inclusão dos agentes econômicos na operação do sistema de potência fornece desafios importantes para o FPO. Apesar de procurar satisfazer conjuntamente o interesse geral da comunidade, as entidades de geração transmissão, distribuição e comercialização de energia elétrica possuem interesses econômicos diferenciados e muitas vezes conflitantes. O operador do sistema requer metodologias específicas para a compatibilização destes interesses, elevando a importância da utilização do FPO. A incorporação dos agentes da rede econômica na rede física impõe mudanças nos conceitos da operação dos sistemas de potência. Extensa bibliografia especializada tem surgido a partir deste problema, sendo impossível sua consideração no

âmbito do presente trabalho. Alguns trabalhos na área são: CHENG, GALIANA, e MCGILLIS (1997), GALIANA e ILIC (1998), ZOBAN e ILIC (1997a e 1997b), ILIC, GALIANA, e FINK (1998), LANKFORD, McCALLEY e SAINI (1996), MUKERJI *et al.* (1992) RAJARAMAN, SALLASHKAR, e ALVARADO (1997) SHIRMOHAMMADI *et al.* (1991) e outros.

2.11. Abrangência do Presente Trabalho.

Neste trabalho, visa-se considerar a robustez da trajetória de convergência e os tempos de cálculo nos métodos de Pontos Interiores Primais-duais para PNL. A robustez na solução e os tempos de cálculo são considerações muitas vezes conflitantes. A melhora na robustez da solução requer em geral cálculos adicionais, com tempos que podem ser consideravelmente maiores que os requeridos para a aplicação direta dos métodos. Os algoritmos de Pontos Interiores permitem a determinação da trajetória central percorrida até a convergência. Regulando a distância da trajetória ao caminho central da região de convergência, os métodos de Pontos Interiores permitem atingir uma solução ótima em regiões limitadas por grande número de restrições de desigualdade. Entretanto, uma centralização forte da trajetória pode requerer um número maior de iterações até a convergência.

A possibilidade da utilização de diferentes critérios para determinar a centralização desejada permite definir as diversas classes de algoritmos de Pontos Interiores. No Capítulo 4 deste trabalho são analisadas algumas destas versões.

2.12. Conclusões.

O FPO permite a consideração num único problema das equações de balanço de potência na rede e das considerações econômicas no despacho do sistema. A modelagem do FPO resulta num problema de otimização não-linear de grande porte, em geral não convexo, que deve ser resolvido via metodologias adequadas. Numerosos métodos têm sido utilizados para resolver este problema, aplicando-se algoritmos de programação linear e não-linear. O método de Pontos Interiores para PNL permite a manipulação das restrições de desigualdade numa forma eficiente, tendo se mostrado competente na resolução do FPO. No capítulo a seguir, o método de Pontos Interiores convencional é formulado.

Capítulo 3.

Formulação Geral do Método de Pontos Interiores Primal-Dual para Programação Não-Linear

3.1. Introdução

Este capítulo inicia a análise do método de Pontos Interiores Primal-Dual para PNL. Uma vez expresso um problema geral de otimização, o método é formalizado a partir da perturbação das condições de complementaridade e da modificação da condição relativa às restrições de desigualdade. As equações não-lineares resultantes são resolvidas via método de Newton. O sistema linear utilizado em cada iteração é de grande dimensão, porém realizando-se uma manipulação matemática divide-se a sua solução em um sistema reduzido e uma série de equações auxiliares.

Adicionalmente, apresenta-se as principais características do sistema reduzido a ser resolvido, com ênfase nas particularidades que podem ser usadas na vetorização da resolução do sistema linear. Outras características do processo são consideradas, como o ponto inicial, os critérios de convergência, etc., formulando-se um algoritmo geral para o método.

Finalmente, é considerada a equivalência entre as duas formulações teóricas do método de Pontos Interiores: aquela baseada no uso de Funções de Barreira Logarítmica e aquela baseada na perturbação das equações de otimalidade de primeira ordem.

3.2. O Problema Original.

Suponha-se um problema geral de otimização, não-linear e restrito, tal como o mostrado em equações (3.1) à (3.3).

$$\text{Min. } f(\mathbf{x}) \quad (3.1)$$

$$\text{s.a. } \mathbf{g}(\mathbf{x}) = \mathbf{0} \quad (3.2)$$

$$\mathbf{h}(\mathbf{x}) \leq \mathbf{0} \quad (3.3)$$

onde, como no Capítulo 2, $f(\mathbf{x})$ é o índice de desempenho ou função objetivo; $\mathbf{g}(\mathbf{x})$ e $\mathbf{h}(\mathbf{x})$ são as restrições de igualdade e desigualdade, respectivamente; e \mathbf{x} são as variáveis do problema de otimização.

As equações (3.2) e (3.3) formam a *região de convergência*, ou seja, o conjunto dos pontos factíveis do problema. A região de convergência é a interseção de dois conjuntos de pontos: a *região de viabilidade das restrições de desigualdade* (conjunto dos pontos \mathbf{x} que satisfazem $\mathbf{h}(\mathbf{x}) \leq \mathbf{0}$) e a *região de viabilidade das restrições de igualdade* (os pontos \mathbf{x} para os quais $\mathbf{g}(\mathbf{x}) = \mathbf{0}$).

Na seção 2.8 do Capítulo 2, são apresentadas as condições de otimalidade de primeira ordem de KKT para este problema. O número de variáveis duais daquela formulação pode ser reduzida, sendo no ponto solução os multiplicadores de complementaridade das restrições de desigualdade iguais aos multiplicadores de Lagrange para as mesmas restrições. Por conseguinte, podem ser representadas estas duas funções numa mesma variável, resultando as condições nas equações (3.4) à (3.8)

$$\nabla f(\mathbf{x}^\circ) + \nabla \mathbf{g}(\mathbf{x}^\circ) * \boldsymbol{\lambda}^\circ + \nabla \mathbf{h}(\mathbf{x}^\circ) * \boldsymbol{\pi}^\circ = \mathbf{0} \quad (3.4)$$

$$\mathbf{g}(\mathbf{x}^\circ) = \mathbf{0} \quad (3.5)$$

$$\mathbf{h}(\mathbf{x}^\circ) \leq \mathbf{0} \quad (3.6)$$

$$[\boldsymbol{\pi}^\circ] * \mathbf{h}(\mathbf{x}^\circ) = \mathbf{0} \quad (3.7)$$

$$\boldsymbol{\pi}^\circ \geq \mathbf{0} \quad (3.8)$$

As equações (3.5) e (3.6) expressam que a solução x^o deve pertencer à região de convergência. Por conseguinte, estas equações estabelecem a *viabilidade primal* do problema. A equação (3.4) fornece as condições que devem ser satisfeitas na solução pelas variáveis duais (λ^o, π^o) , expressando a *viabilidade dual* do problema.

A equação (3.7) representa a complementaridade entre os multiplicadores de Lagrange e o valor no ponto das restrições de desigualdade. Esta condição merece uma consideração especial e possui as seguintes características:

- É não-linear, pois resulta da multiplicação das variáveis duais π por uma função das variáveis primais x ;
- Pode ser entendida como a resolução de um problema combinatório. Com efeito, na solução do problema de otimização, para cada restrição i , $h_i(x^o) = 0$ (se a restrição de desigualdade é ativa na solução) ou π_i é nulo (para restrição não ativa).

A solução de problemas combinatórios pode introduzir dificuldades no processo de convergência.

A equação (3.8) penaliza a inviabilidade das restrições de desigualdade, garantindo que as restrições (3.3) sejam satisfeitas.

As equações (3.4) a (3.8) devem ser satisfeitas no ponto candidato a ótimo. A resolução direta das equações resultantes da aplicação destas condições através de um método iterativo constitui o Método de Newton para FPO. Nas equações (3.4)-(3.8) existe um tratamento diferenciado entre as restrições de igualdade e desigualdade. Requer-se portanto o conhecimento do conjunto de restrições que são ativas no ponto solução. Entretanto, o conjunto de desigualdades ativas no ponto candidato a ótimo pode não ser o mesmo que aquele do ponto inicial, ou do percurso do processo iterativo. Para a solução direta das equações (3.4) à (3.8) requer-se então avaliações no conjunto ativo de restrições de desigualdade e da adição de funções penalidade.

3.3. As Equações de KKT Perturbadas.

A fim de evitar os inconvenientes citados na resolução direta das equações de KKT via método de Newton, o método de Pontos Interiores propõe a modificação das condições de KKT nos seguintes pontos:

3.3.1. Tratamento das Restrições de Desigualdade.

O Método de Pontos Interiores propõe a transformação das restrições de desigualdade em restrições de igualdade, pela utilização de variáveis de folga. Às equações (3.3), são adicionadas as variáveis de folga $s \geq 0$, da forma mostrada na equação (3.9), transformando-as em igualdades.

$$h(x) + s = 0 \quad (3.9)$$

A adição das variáveis primais de folga s modifica a dimensão do problema de otimização, incrementando o número de variáveis. Entretanto, o problema original é transformado num problema de otimização restrito exclusivamente a restrições de igualdade. Com esta metodologia, a utilização de funções de penalidade e a determinação do conjunto de restrições de desigualdade ativas na solução deixam de ser necessárias.

Na solução, a equação (3.9) deve ser satisfeita. Por conseguinte, substituindo na equação (3.7) a expressão $h(x^o) = -s^o$, resulta

$$[\pi^o]^* s^o = 0 \quad (3.10)$$

3.3.2. Tratamento das Equações de Complementaridade.

O Método de Pontos Interiores propõe a relaxação da equação de complementaridade estrita expressa em (3.10) durante o processo iterativo. A condição expressa na equação de complementaridade deve ser satisfeita apenas na solução do problema. Entretanto, no processo iterativo de solução é possível perturbar a equação (3.10) através da adição de um parâmetro $\mu \geq 0$, segundo a equação (3.11).

$$[\pi]^* s = \mu^* e \quad (3.11)$$

onde μ é o parâmetro da perturbação; e e é um vetor unitário da dimensão requerida.

Para cada restrição i , os valores possíveis de π_i e s_i na equação (3.10) correspondem aos



eixos do plano (ou é $\pi_i = 0$ ou $s_i = 0$). Por conseguinte, o processo iterativo deve procurar atingir a solução das equações restantes com uma condição muito estrita. A perturbação da equação de complementaridade (como na equação (3.11)) permite valores de π_i e s_i em todo o plano. Assim, a perturbação na equação de complementaridade pode ser entendida como uma relaxação desta condição no espaço de soluções. O único ponto solução da equação (3.10) é transformado pela perturbação no conjunto de pares da equação (3.12).

$$\chi = \{(s_i, \pi_i) : s_i * \pi_i = c, i = 1, \dots, nd\} \quad (3.12)$$

No ponto candidato ao ótimo, a solução da equação (3.11) deve ser igual à obtida pela expressão original (3.7), razão pela qual no final do processo iterativo deve ser satisfeita a condição $\mu \rightarrow 0$. Desde um valor inicial μ^1 , qualquer seqüência que faz reduzir μ a um valor nulo pode ser utilizada. Entretanto, a forma de determinar o parâmetro da perturbação condiciona a trajetória percorrida para atingir a solução.

3.3.3. Formulação das Equações de KKT Perturbadas.

As modificações descritas nas equações de otimalidade de KKT permitem a formulação de um novo conjunto de equações, denominadas de *Equações de KKT Perturbadas*. As mesmas são expressas nas equações (3.13) à (3.17).

$$\nabla f(x) + \nabla g(x) * \lambda + \nabla h(x) * \pi = 0 \quad (3.13)$$

$$g(x) = 0 \quad (3.14)$$

$$h(x) + s = 0 \quad (3.15)$$

$$[\pi] * s = \mu * e \quad (3.16)$$

$$(s, \pi, \mu) \geq 0 \quad (3.17)$$

As equações de KKT perturbadas são utilizadas no processo de solução do problema de otimização. Cabe salientar que o ponto solução obtido em regiões convexas por estas equações é o mesmo que aquele obtido pela solução das equações de KKT originais. Entretanto, o caminho percorrido para atingir o ponto solução é possivelmente distinto em ambas formulações.

3.4. A Solução Iterativa.

3.4.1. O Método de Newton para Solução de Equações Não-Lineares.

O sistema de equações (3.13)-(3.17) é não-linear e de grande porte. Por conseguinte, é utilizado o método de Newton de resolução de equações não-lineares para obter a sua solução. Como explicitado no Capítulo 2, o método de Newton consiste num processo iterativo, no qual aproxima-se um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$ ao ponto solução $(x^o, s^o, \lambda^o, \pi^o)$ através de uma seqüência de pontos $(x^k, s^k, \lambda^k, \pi^k)$. A seqüência de pontos até a solução é a *trajetória* ou *caminho* percorrido durante o processo iterativo.

Por simplicidade na notação, consideremos as equações a serem resolvidas ((3.13) à (3.17)) representadas como $q(z) = \theta$. O método de Newton é um método de ascendência, com uma direção de busca desde um ponto da trajetória z^k definida pela equação (3.18).

$$\Delta z^k = -(\nabla q(z^k))^{-1} * q(z^k) \quad (3.18)$$

A resolução explícita da inversa de uma matriz não é computacionalmente eficiente. Geralmente, a direção de variação nos parâmetros Δz^k é obtida pela resolução do sistema linear (3.19).

$$\nabla q(z^k) * \Delta z^k = -q(z^k) \quad (3.19)$$

3.4.2. Condições Necessárias para a Utilização do Método de Newton

As condições necessárias para a utilização do método de Newton são (EL-BAKRY *et al.*, 1996):

- a) Existência de um ponto ótimo $(x^o, s^o, \lambda^o, \pi^o)$, solução do sistema de equações (3.1) à (3.3) que satisfaça as condições de KKT (3.4)-(3.8).

- b) Existência e continuidade das segundas derivadas parciais locais no ponto ótimo.

- c) Regularidade: O conjunto das restrições de igualdade e restrições de desigualdade no limite, no ponto ótimo, é linearmente independente.

- d) Suficiência de Segunda Ordem: Para todo $\eta \neq 0$ que satisfaz $\nabla_x h_i(x^o)' * \eta = 0$ e $\nabla_x g_i(x^o)' * \eta = 0$ no ponto ótimo, tem-se $\eta' * \nabla_{xx}^2 L(x^o, s^o, \lambda^o, \pi^o) * \eta > 0$.

- e) Complementaridade Estrita: Para todo i , $\pi_i^o * s_i^o \geq 0$ no ponto ótimo.

As condições teóricas para a utilização do método de Newton geralmente não são avaliadas nas aplicações reais para sistemas de potência de grande porte devido à complexidade de cálculo. Assim, não é garantido que o ponto ótimo obtido seja um ótimo global. Mesmo assim, o processo de otimização fornece, em geral, soluções melhores do que a heurística. A fim de dar continuidade à análise teórica do método de Pontos Interiores, são consideradas satisfeitas as condições para a aplicação do método de Newton para resolução iterativa de equações não-lineares.

3.4.3. Montagem do Sistema Linear Completo.

Substituindo $q(z^k)$ na equação (3.19) pelos seus elementos constituintes, e eliminando, por simplicidade de notação, o supra-índice k , resolve-se em cada iteração do processo de solução um sistema linear da forma expressa em (3.20).

$$\begin{bmatrix} \nabla^2 L(x, s, \lambda, \pi) & \nabla g(x) & \nabla h(x) & 0 \\ \nabla g(x)' & 0 & 0 & 0 \\ \nabla h(x)' & 0 & 0 & [e] \\ 0 & 0 & [s] & [\pi] \end{bmatrix} * \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta \pi \\ \Delta s \end{bmatrix} = - \begin{bmatrix} \nabla L(x, s, \lambda, \pi) \\ g(x) \\ h(x) + s \\ [\pi] * s - \mu * e \end{bmatrix} \quad (3.20)$$

onde:

$$\nabla^2 L(x, s, \lambda, \pi) = \nabla^2 f(x) + \nabla^2 g(x) * \lambda + \nabla^2 h(x) * \pi$$

$$\nabla L(x, s, \lambda, \pi) = \nabla f(x) + \nabla g(x) * \lambda + \nabla h(x) * \pi .$$

A matriz do sistema linear (3.20) possui as seguintes propriedades: a) dimensões elevadas, 10 a 12 vezes a ordem do sistema elétrico; b) alta esparsidade, não só pela existência de numerosas submatrizes nulas, mas também pela esparsidade própria das matrizes não-nulas; c) não-simetria numérica por ser, em geral, $[s] \neq [e]$, sendo $[e]$ uma matriz diagonal unitária; d) simetria na estrutura dos elementos não nulos da matriz.

3.4.4. Redução do Sistema Linear.

O sistema linear (3.20) pode ser reduzido na sua dimensão pela resolução externa de algumas das suas equações. As expressões extraídas do sistema linear são conhecidas como *equações auxiliares*. Além da redução na ordem do sistema, este procedimento permite obter simetria numérica e estrutural na matriz resultante.

Considere-se em primeiro lugar a determinação da direção do incremento das variáveis de folga. Da penúltima linha do sistema linear (3.20), pode-se calcular Δs como expressa a equação (3.21).

$$\Delta s = -(h(x) + s) - \nabla h(x) * \Delta x \quad (3.21)$$

Como se observa em (3.21), Δs pode ser expresso exclusivamente em função dos incrementos Δx .

O incremento nas variáveis duais $\Delta \pi$ é obtido de forma similar. Da última equação de (3.20), pode-se obter $\Delta \pi$ em função de Δs , como expressa (3.22).

$$\Delta \pi = \pi - \left[\frac{1}{s} \right] * \mu * e - \left[\frac{\pi}{s} \right] * \Delta s \quad (3.22)$$

onde $\left[\frac{1}{s} \right]$ e $\left[\frac{\pi}{s} \right]$ são as matrizes diagonais de elementos $\left(\frac{1}{s_{ii}} \right)$ e $\left(\frac{\pi_{ii}}{s_{ii}} \right)$, respectivamente.

Substituindo a equação (3.21) na (3.22), e o resultado na primeira equação da expressão (3.20), obtém-se o sistema linear mostrado em (3.23).

$$\begin{bmatrix} H(x, s, \lambda, \pi) & \nabla g(x) \\ \nabla g(x)' & 0 \end{bmatrix} * \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} t_1(x, s, \lambda, \pi) \\ t_2(x) \end{bmatrix} \quad (3.23)$$

onde

$$H(x, s, \lambda, \pi) = \nabla^2 L(x, s, \lambda, \pi) + \nabla h(x)' * \left[\frac{\pi}{s} \right] * \nabla h(x) \quad (3.24)$$

$$t_1(x, s, \lambda, \pi) = \nabla L(x, s, \lambda, \pi) + \nabla h(x)' * \left\{ \left[\frac{\pi}{s} \right] * h(x) - \left[\frac{1}{s} \right] * \mu * e \right\} \quad (3.25)$$

$$t_2(x) = g(x) \quad (3.26)$$

Em cada iteração, o sistema linear da equação (3.23) deve ser resolvido a fim de se obter o incremento nas variáveis de otimização x e nos multiplicadores de Lagrange das restrições de igualdade λ . Posteriormente, as equações auxiliares (3.21) e (3.22) são resolvidas para se determinar os incrementos nas variáveis de folga s e nos multiplicadores de Lagrange das restrições de desigualdade π .

3.4.5. Características do Sistema Linear Reduzido.

A fim de visualizar melhor a estrutura esparsa da matriz da equação (3.23), ilustra-se na Figura 4 o sistema linear correspondente à resolução de um FPO para o sistema teste IEEE 30 barras. Este sistema foi escolhido porque favorece a visualização da estrutura esparsa, característica que será acentuada na medida em que a dimensão do sistema de potência aumenta.

Pode-se observar que a matriz de coeficientes é composta de três submatrizes principais H , $g(x)$ e $g(x)'$, posteriormente descritas nesta seção, assim como os subvetores do lado direito t_1 e t_2 .

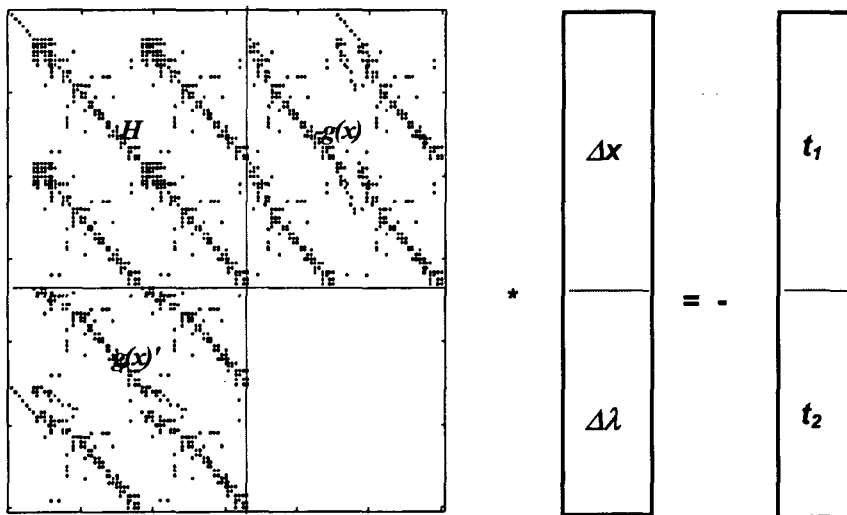


Figura 4: Sistema Linear para IEEE 30 barras.

Na Figura 4, cada ponto da matriz corresponde a um elemento não-nulo. A esparsidade da matriz do sistema linear exemplo é de 90%.

Conforme pode-se observar na Figura 4, a matriz do sistema linear do Método de Pontos Interiores apresenta semelhanças com aquela do método de Newton de otimização. Uma

análise mais detalhada deste sistema linear permite encontrar peculiaridades que devem ser consideradas.

A dimensão do sistema linear a ser resolvido no método de Pontos Interiores é a somatória do número de variáveis primais originais \mathbf{x} mais o número de restrições de igualdade $\mathbf{g}(\mathbf{x})$ originais do problema. Em geral, este sistema linear é mais reduzido que o correspondente ao método de Newton. Neste método, são adicionadas ao sistema linear as restrições de desigualdade ativas na iteração, incrementando o número de equações de igualdade a serem resolvidas no sistema linear. A redução obtida pelo método de Pontos Interiores na dimensão do sistema linear é obtida pela extração das equações auxiliares do sistema linear. Em geral, este procedimento é mais eficiente computacionalmente que a resolução de um sistema linear de grandes dimensões.

Além da dimensão, existem diferenças importantes entre os sistemas lineares dos métodos de Newton e de Pontos Interiores. Algumas destas características são enunciadas a seguir, na análise das matrizes e vetores envolvidos.

3.4.5.1. Matriz Hessiana (H)

No método de Newton, \mathbf{H} é a submatriz de segundas derivadas da função Lagrangeana com respeito às variáveis primais. Esta matriz possui elementos não-nulos na mesma disposição que a matriz de admitâncias \mathbf{Y} , é quadrada e tem a mesma dimensão que o número de variáveis \mathbf{x} .

No método de Pontos Interiores, a submatriz \mathbf{H} é modificada pela adição do termo

$\nabla \mathbf{h}(\mathbf{x})' * \left[\frac{\pi}{s} \right] * \nabla \mathbf{h}(\mathbf{x})$ (equação (3.24)). A matriz $\nabla \mathbf{h}(\mathbf{x})$, derivada das restrições de

desigualdade em relação às variáveis \mathbf{x} , é retangular e de dimensão $(n \times nd)$. Em geral, a

multiplicação desta matriz pela sua transposta (afetadas pela matriz diagonal $\left[\frac{\pi}{s} \right]$)

ocasiona o surgimento de elementos não-nulos em posições distintas às originais de \mathbf{H} .

Por conseguinte, a matriz \mathbf{H} do método de Pontos Interiores é em geral menos esparsa que a correspondente ao método de Newton.

Em relação à Figura 4, também se pode observar que a matriz \mathbf{H} apresenta três campos densamente ocupados, na vizinhança da diagonal principal e de duas diagonais simétricas posicionalmente. Deve ainda ser mencionada a disparidade numérica entre os elementos da referida matriz e sua diagonal não-nula, características a serem consideradas nos procedimentos de fatoração.

3.4.5.2. Matriz Jacobiana ($\mathbf{g}(\mathbf{x})$)

A matriz Jacobiana corresponde às primeiras derivadas parciais das equações de igualdade com respeito às variáveis \mathbf{x} . É pois uma matriz retangular, de número de linhas igual ao número de variáveis e número de colunas igual ao número de restrições de igualdade originais do problema.

É interessante notar a particular concentração dos elementos não-nulos em linhas oblíquas, dominante nos elementos da matriz Jacobiana, conforme mostrado na Figura 4. Esta matriz provém da união de blocos diagonal-dominantes, resultando numa configuração a ser explorada posteriormente na implementação de técnicas computacionais de Alto Desempenho.

3.4.5.3. Vetor Independente

Conforme observado na Figura 4, o vetor independente é formado por duas parcelas. Similarmente ao que acontece com a matriz \mathbf{H} , a parcela superior \mathbf{t}_1 é distinta nos métodos de Newton e de Pontos Interiores. No primeiro, \mathbf{t}_1 é o gradiente da função Lagrangeana em relação às variáveis \mathbf{x} . No método de Pontos Interiores, a este vetor é

adicionado o vetor $\nabla h(\mathbf{x})' * \left\{ \left[\frac{\pi}{s} \right] * h(\mathbf{x}) - \left[\frac{1}{s} \right] * \mu * e \right\}$. A parcela inferior \mathbf{t}_2 , derivada das

restrições de igualdade em relação às variáveis \mathbf{x} , é similar nos métodos de Newton e de Pontos Interiores. A dimensão do vetor \mathbf{t}_2 pode resultar maior no método de Newton, se existirem restrições de desigualdade ativas na iteração.

3.4.6. Determinação dos Passos e Atualização das Variáveis.

A equação (3.17) determina a não-negatividade das variáveis \mathbf{s} e π na trajetória ao ponto ótimo. Neste trabalho, utilizam-se dois passos, correspondentes respectivamente às variáveis primais \mathbf{s} e duais π . Eles são calculados segundo as expressões (3.27) e (3.28) (EL-BAKRY *et al*, 1996).

$$\alpha_p = \min \left[\min_{\Delta s_j < 0} \frac{s_j}{|\Delta s_j|}, 1.0 \right] \quad (3.27)$$

$$\alpha_d = \min \left[\min_{\Delta\pi_j < 0} \frac{\pi_j}{|\Delta\pi_j|}, 1.0 \right] \quad (3.28)$$

Calculados os fatores de passo, atualiza-se as variáveis primais e duais do problema, determinando o próximo ponto na trajetória (equações (3.29) à (3.32)).

$$x = x + \kappa * \alpha_p * \Delta x \quad (3.29)$$

$$s = s + \kappa * \alpha_p * \Delta s \quad (3.30)$$

$$\lambda = \lambda + \kappa * \alpha_d * \Delta \lambda \quad (3.31)$$

$$\pi = \pi + \kappa * \alpha_d * \Delta \pi \quad (3.32)$$

onde κ é o fator de redução dos passos.

Para $\alpha_p < 1.0$, a equação (3.27) determina o passo primal máximo para que pelo menos um componente do vetor s seja nulo. O mesmo acontece na equação (3.28) com a determinação do passo dual em relação às variáveis π . Como o objetivo de garantir a não-negatividade das variáveis e evitar problemas numéricos, é conveniente a redução dos passos por um fator κ . Um valor usado na prática é $\kappa = 0.99995$.

3.5. Convergência do Processo Iterativo.

As equações de otimalidade perturbadas (3.13) à (3.17) devem ser satisfeitas no ponto solução $(x^o, s^o, \lambda^o, \pi^o)$. Além destas condições, é necessário que na convergência o parâmetro $\mu \rightarrow 0$ para aproximar as equações perturbadas e as originais (3.4) à (3.8).

Não existem muitos trabalhos que provam a convergência de um método de Pontos Interiores em PNL, nem sequer para problemas convexos. Na referência EL-BAKRY *et al.* (1996) é demonstrada a convergência de um algoritmo de Pontos Interiores similar ao aqui apresentado. Nesse artigo, os autores apresentam inicialmente uma formulação linear do método, estendendo a explicação para o caso não-linear. A convergência do processo é demonstrada global e localmente com o uso da função de mérito $\beta(x, s, \lambda, \pi)$ (equação (3.33)).

$$\beta(x, s, \lambda, \pi) = \|\nabla_x L(x, s, \lambda, \pi)\|_2^2 + \|g(x)\|_2^2 + \|h(x) + s\|_2^2 + \|\pi\|_2^2 \quad (3.33)$$

A função de mérito da equação (3.33) é avaliada em cada iteração, e representa a norma euclidiana dos resíduos das equações de KKT.

EL-BAKRY *et al.* (1996) demonstram que, para uma apropriada eleição da seqüência de parâmetros $\{\mu^k\}$, existe uma sucessão de passos primais-duais $\{\alpha^k\}$ para os quais $\beta(x, s, \lambda, \pi)$ decresce monotonicamente e as iterações convergem a um ponto $(x^o, s^o, \lambda^o, \pi^o)$ no qual $\beta(x^o, s^o, \lambda^o, \pi^o) = 0$.

A convergência é demonstrada para a função de mérito da equação (3.33). Como esta função é baseada nas condições de otimalidade de primeira ordem, o algoritmo pode convergir a pontos de sela ou máximos locais em problemas matemáticos complexos. SHANNO e SIMANTIRAKI, citados em VANDERBEI e SHANNO (1991), mostram problemas testes nos quais o algoritmo converge a pontos não-mínimos, inclusive a aqueles nos quais o Jacobiano $\nabla g(x)$ se torna singular. A demonstração de convergência com funções de mérito mais gerais não é conhecida pelo autor deste texto até esta data.

As características do FPO permitem em geral a convergência dos métodos de Pontos Interiores a mínimos. A dificuldade no tratamento das equações não-lineares impede saber se estes pontos são mínimos locais ou globais.

3.6. O Ponto Inicial.

A solução do problema de otimização deve-se encontrar no interior ou na fronteira da região de convergência. O ponto inicial e a trajetória de convergência não possuem essa restrição. De fato, as condições de otimalidade (3.13) à (3.16) são referidas ao ponto candidato ao ótimo, não sendo necessariamente satisfeitas durante o processo iterativo. A resolução iterativa determina uma seqüência de pontos que aproximam o ponto inicial à solução. Em particular, as equações que determinam a inclusão do ponto na região de convergência ((3.14) e (3.15)) serão em geral satisfeitas apenas no final do processo iterativo. Esta é uma importante vantagem na utilização das versões dos métodos de Pontos Interiores para PNL. Os algoritmos lineares de Pontos Interiores requerem uma trajetória interior à região formada pelas restrições de desigualdade. Por conseguinte, esses algoritmos necessitam da determinação de um ponto inicial interior a esta região. A

utilização das condições de otimalidade de KKT na forma expressa nas equações (3.13) à (3.17) permite eliminar esta restrição. Entretanto, continuam existindo limitações para alguns dos valores iniciais das variáveis do problema. Na determinação do ponto inicial, existem dois grupos de variáveis:

- Variáveis livres: as variáveis originais do problema x e os multiplicadores de Lagrange das restrições de igualdade λ podem assumir qualquer valor inicial.
- Variáveis não-negativas: as variáveis de folga s , os multiplicadores de Lagrange correspondentes às restrições de desigualdade π e o parâmetro da perturbação μ devem ser não-negativos durante o processo iterativo e, em particular, no ponto inicial (equação (3.17)).

A não-negatividade do parâmetro μ é garantida no seu cálculo. Partindo de um ponto inicial com variáveis s e π não-negativas, esta condição é mantida no processo iterativo mediante a utilização de passos, como mostrado na Seção 3.4.6.

3.7. Algoritmo Geral dos Métodos de Pontos Interiores para PNL.

Os métodos de Pontos Interiores em PNL são sintetizados no seguinte algoritmo:

Algoritmo 3-1: Algoritmo Geral dos Métodos de Pontos Interiores para PNL.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$. Seja $\mu^l > 0$.
 2. Montagem e ordenação simbólica do sistema linear (3.23).
 3. Resolução do sistema linear (3.23) e as equações auxiliares (3.21) e (3.22).
 4. Cálculo dos fatores de passo através de (3.27) e (3.28).
 5. Atualização das variáveis via equações (3.29) à (3.32).
 6. Cálculo do parâmetro μ .
 7. Teste de Convergência: Convergência: FIM; Não-convergência: CONTINUA.
 8. Atualização do sistema linear (3.23).
 9. Retornar ao passo 3.
-

O Algoritmo 3-1 é uma versão mais completa do algoritmo de Pontos Interiores apresentado no Capítulo 2.

3.8. O Parâmetro da Perturbação μ

Como expresso anteriormente, o parâmetro da perturbação μ deve tender a zero na solução do processo iterativo. Assim, pode ser utilizada qualquer seqüência tal como a mostrada em (3.34).

$$\beta = \{(\mu^1, \dots, \mu^k, \dots, \mu^o) : \mu^o < \mu^k, \mu^o < \delta\} \quad (3.34)$$

onde δ é um número suficientemente pequeno.

Nos métodos Primais-duais, a equação (3.16) fornece uma forma de cálculo do parâmetro μ . Na solução, o produto de cada variável de folga s_i vezes o multiplicador de Lagrange correspondente π_i deve ser nulo, como expressa a equação de KKT original (3.7). No processo iterativo, o produto $\mu_i = s_i * \pi_i$ fornece uma medida da distância do ponto corrente ao ponto candidato a ótimo, expressa em função da equação de complementaridade (3.16). Esta medida é distinta para cada desigualdade. Na prática, é utilizado um único parâmetro μ para todas as restrições $h(x)$.

Suponha-se um parâmetro único $\bar{\mu}$ que expresse a distância entre o ponto atual e a solução. EL-BAKRY *et al.* (1996) utilizam como medida desta distância o mínimo produto $s_i * \pi_i$, calculando $\bar{\mu}$ como na expressão (3.35).

$$\bar{\mu} = \min([\pi] * s) \quad (3.35)$$

A maioria dos trabalhos na área (GRANVILLE, 1994, WRIGHT, 1997 e outros) utiliza como distância a média dos produtos $s_i * \pi_i$ em todas as restrições de desigualdade. Assim, $\bar{\mu}$ fica expresso como na equação (3.36).

$$\bar{\mu} = \frac{\pi'^* s}{nd} \quad (3.36)$$

onde, como anteriormente, nd é o número de restrições de desigualdade $h(x)$.

O valor $\bar{\mu}$ expressa o valor da distância entre o ponto da iteração presente e a solução. Para que $\mu \rightarrow 0$ no processo iterativo, deve-se utilizar um parâmetro menor que o atual na iteração seguinte. Assim, o valor de μ é calculado em função da distância $\bar{\mu}$ reduzida num fator $0 < \sigma < 1$, como na equação (3.37).

$$\mu = \sigma * \bar{\mu} \quad (3.37)$$

onde σ é o *parâmetro de combinação das direções*.

O parâmetro σ modifica a direção de busca utilizada em cada iteração. Como será visto posteriormente, este parâmetro é parte importante na diferenciação entre as versões do Método de Pontos Interiores Primal-Dual de PNL.

3.8.1. O Método de Pontos Interiores Primal-Dual Convencional para PNL.

O método de Pontos Interiores Primal-Dual convencional em PNL é baseado no Algoritmo 3-1. A redução do parâmetro da perturbação μ para este algoritmo é realizada como explicado anteriormente, com um parâmetro σ constante. O valor ótimo de σ (aquele que reduz o número de iterações do processo iterativo) depende do sistema considerado e das condições iniciais do problema. Pelas características não-lineares do problema de otimização, não se conhecem métodos que permitam a determinação "a priori" de um σ único que redunde num tempo mínimo de cálculo até a solução. Um valor muito utilizado na prática é $\sigma = 0.1$. Por conseguinte, em cada iteração do método de Pontos Interiores Convencional μ é calculado como na equação (3.38).

$$\mu = 0.1 * \frac{\pi'^* s}{nd} \quad (3.38)$$

O método de Pontos Interiores Primal-Dual Convencional é muito utilizado em trabalhos na área de FPO. Por isso, é utilizado como referência na comparação com outras versões de Pontos Interiores.

3.9. Formulação do Método de Pontos Interiores através da Barreira Logarítmica.

O método de pontos Interiores pode ser formulado através da utilização de barreiras logarítmicas. Nesta seção é analisada esta formulação, estudando-se algumas das suas características.

As equações de otimalidade de primeira ordem (3.4) à (3.8) devem ser satisfeitas no ponto ótimo. Para melhor compreensão, elas são repetidas novamente nesta seção.

$$\nabla f(\mathbf{x}^o) + \nabla g(\mathbf{x}^o) * \lambda^o + \nabla h(\mathbf{x}^o) * \pi^o = \mathbf{0} \quad (3.39)$$

$$g(\mathbf{x}^o) = \mathbf{0} \quad (3.40)$$

$$h(\mathbf{x}^o) \leq \mathbf{0} \quad (3.41)$$

$$[\pi^o] * h(\mathbf{x}^o) = \mathbf{0} \quad (3.42)$$

$$\pi^o \geq \mathbf{0} \quad (3.43)$$

Suponha-se que em torno da solução $(\mathbf{x}^o, \lambda^o, \pi^o)$ existe uma vizinhança infinitesimal na qual as restrições de viabilidade primal (3.40) e (3.41) são satisfeitas. Considere-se um ponto $(\mathbf{x}, \lambda, \pi)$, pertencente à região factível, infinitesimalmente próximo da solução $(\mathbf{x}^o, \lambda^o, \pi^o)$. Suponha-se ainda que neste ponto, de perturbação da solução ótima, podem ser encontrados multiplicadores λ e $\pi \geq \mathbf{0}$, não simultaneamente nulos, tal que a equação (3.39) seja satisfeita (SANCHEZ, 1983, CASTRONUOVO, 1997). Ou seja, estando o ponto $(\mathbf{x}, \lambda, \pi)$ próximo da solução ótima, o gradiente do Lagrangeano encontra-se ainda no interior do cone convexo constituído pelos gradientes negativos das restrições, e pode ser expresso como uma combinação linear dos mesmos.

Sendo assim definidos os multiplicadores de Lagrange π , em geral não se pode garantir que as equações de complementaridade sejam satisfeitas. As condições de primeira ordem para o ponto perturbado (x, λ, π) são expressas pelas equações (3.44) à (3.48).

$$\nabla f(x) + \nabla g(x) * \lambda + \nabla h(x) * \pi = 0 \quad (3.44)$$

$$g(x) = 0 \quad (3.45)$$

$$h(x) \leq 0 \quad (3.46)$$

$$[\pi] * h(x) \leq 0 \quad (3.47)$$

$$\pi \geq 0 \quad (3.48)$$

Se o ponto (x, λ, π) pertence à região viável, as equações (3.45) e (3.46) devem ser satisfeitas. Entretanto, neste ponto a equação de igualdade (3.42) é transformada na desigualdade (3.47). Esta desigualdade pode ser novamente transformada em igualdade pela adição do vetor de variáveis de folga $\mu > 0$. O sistema de equações resulta nas equações (3.49) à (3.54).

$$\nabla f(x) + \nabla g(x) * \lambda + \nabla h(x) * \pi = 0 \quad (3.49)$$

$$g(x) = 0 \quad (3.50)$$

$$h(x) \leq 0 \quad (3.51)$$

$$[\pi] * h(x) + \mu = 0 \quad (3.52)$$

$$\pi \geq 0 \quad (3.53)$$

$$\mu \geq 0 \quad (3.54)$$

Observa-se que, nas equações (3.52), existe um parâmetro distinto μ_i para cada restrição de desigualdade i .

Através de uma manipulação matemática, podem ser eliminados os multiplicadores π do sistema de equações, substituindo a equação (3.52) na expressão (3.49), resultando o sistema (3.55)-(3.58).

$$\nabla f(x) + \nabla g(x) * \lambda + \nabla h(x) * \left[\frac{\mu}{(-h(x))} \right] * e = 0 \quad (3.55)$$

$$g(x) = 0 \quad (3.56)$$

$$h(x) \leq 0 \quad (3.57)$$

$$\mu \geq 0 \quad (3.58)$$

onde $\left[\frac{\mu}{(-h(x))} \right]$ é a matriz diagonal de elementos $\left(\frac{\mu_i}{(-h_i(x))} \right)$; e $\left(\left[\frac{\mu}{(-h(x))} \right] * e \right)$

define o vetor $\left\{ \frac{\mu_1}{(-h_1(x))}, \dots, \frac{\mu_i}{(-h_i(x))}, \dots, \frac{\mu_{nd}}{(-h_{nd}(x))} \right\}$.

Entretanto, as equações (3.55) à (3.58) representam as condições de otimalidade de primeira ordem para um problema como o representado na equação (3.59).

$$P(x, \lambda, \pi) = f(x) + g(x) * \lambda - \sum_{i=1}^{nd} (\mu_i * \ln(h(x))) \quad (3.59)$$

O gradiente da equação (3.59) é representado pela equação (3.55), e se anula no ponto (x, λ, π) ; isto é, a condição necessária de primeira ordem para este ponto ser mínimo local é satisfeita. Assim, a inclusão da barreira logarítmica na expressão da função objetivo, para $(\mu, s) \geq 0$, permite satisfazer no ponto solução a equação (3.55).

A função $\sum_{i=1}^{nd} (\mu_i * \ln(s_i))$ é chamada *barreira logarítmica*. A idéia de sua utilização é o estabelecimento de uma "penalidade" (como expressa no Capítulo 2), à qual é submetida a função objetivo $f(x)$ a fim de garantir que o ponto solução (x^o, λ^o, π^o) pertença ao interior da região viável formada pelas restrições de desigualdade. A função logarítmica penaliza as trajetórias externas à região de convergência, aproximando a trajetória de busca a um caminho central às restrições de desigualdade.

Similar ao tratamento da equação (3.47), a equação de desigualdade (3.57) pode ser transformada em igualdade através da adição de variáveis de folga $s \geq 0$, segundo a equação (3.60)

$$h(x) + s = 0 \quad (3.60)$$

O sistema de equações resultante é representado nas equações (3.61) à (3.64).

$$\nabla f(x) + \nabla g(x) * \lambda + \nabla h(x) * \begin{bmatrix} \mu \\ s \end{bmatrix} * e = 0 \quad (3.61)$$

$$g(x) = 0 \quad (3.62)$$

$$h(x) + s = 0 \quad (3.63)$$

$$(\mu, s) \geq 0 \quad (3.64)$$

A interpretação do método de Pontos Interiores através de barreiras logarítmicas nas restrições de desigualdade fortalece a imposição da condição de diminuir o parâmetro μ durante o processo iterativo. Se μ é diminuído, o "peso" do termo de penalidade também diminui em relação à função objetivo, permitindo a minimização da função $f(x)$. Um valor constante de μ durante todo o processo iterativo aproximaria a solução ao ponto central da região formada pelas restrições de desigualdade, impedindo a convergência do processo iterativo pela não satisfação das condições de KKT do problema original.

A formulação do método de Pontos Interiores via funções de penalidade tem a vantagem de aproximar a interpretação do mesmo a do método de Newton. As funções de penalidade têm sido usadas desde a sua proposição nos anos 60 (FIACCO e McCORMICK, 1968), sendo mais conhecidas que os métodos de perturbação (ELBAKRY *et al.*, 1996). Entretanto, a formulação via funções de penalidade das restrições de desigualdade pode trazer problemas, tal como acontece no método de Newton.

O vetor gradiente da função (3.59) é representado na equação (3.65).

$$\nabla P(x, \lambda, \pi) = \nabla f(x) + \nabla g(x) * \lambda + \nabla h(x) * \begin{bmatrix} \mu \\ s \end{bmatrix} * e \quad (3.65)$$

Para as restrições $h(x) \leq 0$ ativas na solução (ou seja, satisfeitas na igualdade $h_i(x) = 0$), a variável de folga correspondente é nula ($s_i = 0$). Para este conjunto de restrições ativas na solução, na equação (3.65), $s_i \rightarrow 0$. Entretanto, como estas restrições de desigualdade ativas são restrições de igualdade nesse ponto, a condição $\left(\frac{\mu_i}{s_i}\right) \rightarrow \pi_i$ deve ser satisfeita, com π_i representando os multiplicadores de Lagrange

para estas novas igualdades. Os multiplicadores de Lagrange π_i devem ter valores finitos, e por conseguinte $\mu_i \rightarrow +\infty$. Como acontece com os métodos de penalidade analisados na Seção 2.7, estas condições originam matrizes lineares mal condicionadas, devendo-se realizar manipulações matemáticas (como a utilização da função Lagrangeana aumentada) a fim de eliminar o mau-condicionamento do sistema.

Ambas formulações (via parâmetro de perturbação ou funções de penalidade) chegam às mesmas equações (3.13) à (3.17).

3.10. Conclusões

O método de Pontos Interiores apresenta uma formulação matemática simples, que permite a manipulação de restrições de desigualdade em problemas de grandes dimensões. Na versão não-linear, o ponto inicial para a convergência pode ser encontrado facilmente e sem cálculos adicionais.

Apesar da diminuição da dimensão do sistema linear a ser resolvido em cada iteração, pela exclusão das variáveis de folga e seus respectivos multiplicadores, a ordem deste sistema continua elevada. A resolução do sistema linear consome uma percentagem elevada dos tempos computacionais por iteração, justificando a utilização do Processamento de Alto Desempenho.

A perturbação das equações de otimalidade de primeira ordem introduz o único parâmetro do algoritmo. No Capítulo 4 são mostradas as versões do método de Pontos Interiores, a partir da análise da direção de busca da metodologia.

Capítulo 4.

Versões do Método de Pontos Interiores

4.1. Introdução

O presente capítulo é iniciado com a interpretação dos parâmetros utilizados no método de Pontos Interiores, apresentando a direção de busca utilizada em cada iteração como uma combinação linear de três direções: a direção de Viabilidade das Restrições de Igualdade, a de Centralização e a Afim-Escala. É também formulado o conceito de Caminho Central na região de convergência das restrições de desigualdade, percurso de referência para a trajetória utilizada nos métodos de Pontos Interiores. Os conceitos anteriores são apresentados numa forma gráfica, a fim de facilitar a interpretação geométrica das direções utilizadas.

O método de Pontos Interiores tem evoluído desde a sua formulação original, resultando em diversas versões do algoritmo convencional. Estas variantes têm sido formuladas na área da Otimização Matemática, geralmente para aplicação em problemas de otimização linear. Além do algoritmo convencional apresentado no Capítulo 3, são analisadas aqui sete versões do método de Pontos Interiores, contemplando-se sua adaptação aos problemas não-lineares. A fim de melhorar a aplicabilidade dos métodos considerados, para cada versão é formulado um algoritmo de utilização e são fornecidas algumas considerações para sua implementação.

4.2. Os Parâmetros μ e σ .

As equações de otimalidade de primeira ordem perturbadas, a serem resolvidas no processo iterativo do método de Pontos Interiores, são (eq. (4.1) à (4.5)):

$$\nabla f(\mathbf{x}) + \nabla g(\mathbf{x}) * \lambda + \nabla h(\mathbf{x}) * \pi = 0 \quad (4.1)$$

$$g(\mathbf{x}) = 0 \quad (4.2)$$

$$h(\mathbf{x}) + s = 0 \quad (4.3)$$

$$[\pi] * s = \mu * e \quad (4.4)$$

$$(s, \pi, \mu) \geq 0 \quad (4.5)$$

A solução das equações (4.1) à (4.5) é obtida através de um processo iterativo via método de Newton. Em cada iteração do método de Pontos Interiores é resolvido um sistema linear e equações auxiliares, como mostrado no Capítulo 3. Visando facilitar a compreensão dos conceitos apresentados a seguir, essas equações são repetidas neste capítulo.

$$\begin{bmatrix} H(\mathbf{x}, s, \lambda, \pi) & \nabla g(\mathbf{x}) \\ \nabla g(\mathbf{x})' & 0 \end{bmatrix} * \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} t_1(\mathbf{x}, s, \lambda, \pi) \\ t_2(\mathbf{x}) \end{bmatrix} \quad (4.6)$$

$$\Delta s = -(h(\mathbf{x}) + s) - \nabla h(\mathbf{x}) * \Delta \mathbf{x} \quad (4.7)$$

$$\Delta \pi = -\pi + \left[\frac{1}{s} \right] * \mu * e - \left[\frac{\pi}{s} \right] * \Delta s \quad (4.8)$$

onde

$$H(\mathbf{x}, s, \lambda, \pi) = \nabla^2 L(\mathbf{x}, s, \lambda, \pi) + \nabla h(\mathbf{x})' * \left[\frac{\pi}{s} \right] * \nabla h(\mathbf{x}) \quad (4.9)$$

$$t_1(\mathbf{x}, s, \lambda, \pi) = \nabla L(\mathbf{x}, s, \lambda, \pi) + \nabla h(\mathbf{x})' * \left\{ \left[\frac{\pi}{s} \right] * h(\mathbf{x}) + \left[\frac{1}{s} \right] * \mu * e \right\} \quad (4.10)$$

$$t_2(x) = g(x) \quad (4.11)$$

O parâmetro da perturbação é calculado segundo a expressão (4.12).

$$\mu = \sigma * \frac{\pi'^* s}{nd} \quad (4.12)$$

Define-se *distância primal-dual* numa restrição de desigualdade i , como o produto da expressão primal $\underline{h}_i(x)$ pelo valor da variável dual correspondente π_i . Desde que, na convergência, $s = -h(x)$ (equação (4.3)), freqüentemente formula-se a distância primal-dual em função das variáveis de folga s , procedimento adotado neste trabalho.

A equação (4.12) expressa o cálculo do parâmetro de perturbação como a distância média primal-dual (*dmpd*) de todas as restrições de desigualdade do problema ao ponto candidato a ótimo na iteração corrente, ajustadas pelo parâmetro σ , como na equação (4.13).

$$\mu = \sigma * dmpd \quad (4.13)$$

onde, $dmpd = \frac{\pi'^* s}{nd}$ é o valor da média das distâncias primais-duais nas restrições de desigualdade numa iteração.

Considerando-se o parâmetro $0 < \sigma < 1$, a média das distâncias primais-duais *dmpd* diminui a cada iteração, conforme demonstrado em (EL-BAKRY *et al.*, 1996). Na convergência, para um valor de σ adequado, são satisfeitas as condições de otimalidade do problema original sem perturbação ($\mu \approx 0$), o que indica que o ponto candidato a ótimo é alcançado. Entretanto, a equação (4.12) não é a única expressão possível da variação de μ . O presente trabalho propõe o estudo de formas alternativas de cálculo para o parâmetro σ , visando sua aplicação na solução do FPO. A seguir é realizada uma análise mais detalhada deste parâmetro.

4.3. As Direções de Centralização e Afim-Escala.

A expressão (4.12), utilizada para o cálculo de μ , contém o parâmetro de combinação das direções σ , com limites $0 \leq \sigma \leq 1$. Duas escolhas de σ são de particular interesse:

- $\sigma = 0$: define a direção *Afim-Escala*;
- $\sigma = 1$: define a direção de *Centralização*.

Suponha-se que é utilizada a direção Afim-Escala na resolução do sistema não-linear (4.1)-(4.5). O parâmetro $\sigma = 0$ anula a perturbação μ e o método de Pontos Interiores resulta similar ao método de Newton para FPO, de acordo com o apresentado no Capítulo 2. A única diferença entre essas metodologias é a utilização de variáveis de folga para transformar as restrições de desigualdade em igualdades e a consideração de todas as restrições de desigualdade. Assim, o uso do passo Afim-Escala implica em atingir o ponto ótimo através da resolução de um sistema não-linear de equações com características de convergência similares ao método de Newton para FPO (EL-BAKRY *et al.*, 1996).

Suponha-se, agora, que o passo de centralização é utilizado para a solução das equações (4.1) à (4.5). Neste caso, não existe uma diminuição apreciável de μ durante as iterações, resolvendo-se um problema "perturbado", distinto do original. Na convergência, o método atinge um ponto solução distinto do ótimo, equidistante das restrições de desigualdade, a uma distância primal-dual igual ao valor de μ . Então, o passo de centralização procura atingir um ponto central a uma distancia das restrições de desigualdade dada pela média das distâncias primais-duais (*dmpd*) no ponto inicial.

O método de Pontos Interiores convencional propõe a especificação prévia do valor de σ entre os dois limites considerados, mantendo este valor constante ao longo das iterações.

A seguir é estudado o comportamento das direções Afim-Escala e de Centralização no sistema linear (4.6), e é mostrado que o parâmetro σ realiza uma combinação das direções Afim-Escala e de Centralização.

A aplicação do método de Newton para a resolução da equação não-linear (4.4), resulta na equação (4.14).

$$[s]^* \Delta\pi + [\pi]^* \Delta s = -[\pi]^* s + \mu^* e \quad (4.14)$$

Substituindo μ na equação (4.14) pela expressão (4.13) obtém-se a equação (4.15).

$$[s]^* \Delta\pi + [\pi]^* \Delta s = -[\pi]^* s + \sigma^* \text{dmpd}^* e \quad (4.15)$$

Deve-se destacar, que esta é uma das equações do sistema linear a ser resolvido a cada iteração. Para obter os valores das variáveis $\Delta\pi$ e Δs , é necessária a solução do sistema linear completo e das equações auxiliares, como expressam as equações (4.6) à (4.8).

Numa dada iteração, as direções Afim-Escala e de Centralização exigem o cumprimento das seguintes condições, segundo (4.15):

Para Direção Afim-Escala ($\sigma = 0$):

$$[s]^* \Delta\pi^{ae} + [\pi]^* \Delta s^{ae} = -[\pi]^* s \quad (4.16)$$

Para Direção de Centralização ($\sigma = 1$):

$$[s]^* \Delta\pi^c + [\pi]^* \Delta s^c = -[\pi]^* s + \text{vmpd}^* e \quad (4.17)$$

onde, $\Delta\pi^{ae}$ e Δs^{ae} representam a direção Afim-Escala; e $\Delta\pi^c$ e Δs^c são a direção de Centralização.

Multiplicando (4.17) por σ e somando o resultado à expressão (4.16), obtém-se a equação (4.18).

$$[s]^* \{ \sigma^* \Delta\pi^c + \Delta\pi^{ae} \} + [\pi]^* \{ \sigma^* \Delta s^c + \Delta s^{ae} \} = -[\pi]^* s - \sigma^* [\pi]^* s + \sigma^* \text{vmpd}^* e \quad (4.18)$$

Multiplicando ambos os termos da equação (4.16) por σ , obtém-se o produto $(\sigma * [\pi] * s)$ dado pela equação (4.19).

$$-\sigma * [\pi] * s = \sigma * [s] * \Delta\pi^{ae} + \sigma * [\pi] * \Delta s^{ae} \quad (4.19)$$

Substituindo a equação (4.19) no segundo termo da equação (4.18), e reordenando os termos da equação resultante, obtém-se a expressão (4.20).

$$[s] * \{ \sigma * \Delta\pi^c + (1-\sigma) * \Delta\pi^{ae} \} + [\pi] * \{ \sigma * \Delta s^c + (1-\sigma) * \Delta s^{ae} \} = -[\pi] * s + \sigma * vmpd * e \quad (4.20)$$

Da comparação das expressões (4.15) e (4.20) resultam as equações (4.21)-(4.22).

$$\Delta\pi = \sigma * \Delta\pi^c + (1-\sigma) * \Delta\pi^{ae} \quad (4.21)$$

$$\Delta s = \sigma * \Delta s^c + (1-\sigma) * \Delta s^{ae} \quad (4.22)$$

A análise das expressões (4.21)-(4.22) indica que a direção do método de Pontos Interiores utiliza uma combinação linear em σ , das direções Afim-Escala e de Centralização anteriormente mencionadas. Por esta razão, o parâmetro σ recebe o nome de parâmetro de combinação das direções (VILLELA, 1997).

4.4. Interpretação Gráfica das Direções.

Em cada iteração k do método de Pontos Interiores, procura-se encontrar uma direção $(\Delta x^k, \Delta s^k, \Delta \lambda^k, \Delta \pi^k)$ que aproxime o ponto atual $(x^k, s^k, \lambda^k, \pi^k)$ à solução $(x^o, s^o, \lambda^o, \pi^o)$. Nesta seção, pretende-se interpretar as componentes da direção em função das restrições que devem ser satisfeitas. No ponto solução $(x^o, s^o, \lambda^o, \pi^o)$, as

equações de otimalidade (4.1) a (4.5) devem ser satisfeitas, com $\mu \rightarrow 0$. Em primeiro lugar, é analisada a *Direção de Viabilidade das Restrições de Igualdade*, considerando-se posteriormente as direções de Centralização e Afim-Escala

4.4.1. Representação da Direção de Viabilidade das Restrições de Igualdade.

A equação (4.2) expressa que, na solução, as restrições de igualdade $g(x) = 0$ devem ser satisfeitas. Seja \mathcal{G} o conjunto de pontos para os quais as restrições de igualdade são satisfeitas, segundo (4.23).

$$\mathcal{G} = \{x \in \mathbb{R}^n : g(x) = 0\} \quad (4.23)$$

Em cada iteração do método de Pontos Interiores é realizada uma linearização das condições de KKT perturbadas, visando determinar uma direção que satisfaça ditas condições. Para as restrições de igualdade, esta direção $(\Delta x^k, \Delta s^k, \Delta \lambda^k, \Delta \pi^k)$ é determinada pela aplicação do método de Newton na equação (4.2), resultando na expressão (4.24).

$$\Delta x^{vi} = -(\nabla g(x))^{-1} * g(x) \quad (4.24)$$

onde Δx^{vi} é uma componente da direção $(\Delta x^k, \Delta s^k, \Delta \lambda^k, \Delta \pi^k)$ devida às restrições de igualdade.

A equação (4.24) não pode ser resolvida independentemente do sistema linear (4.6). A direção $(\Delta x^{vi}, 0, 0, 0)$ fornece somente o componente da direção de busca que leva o ponto $(x^k, s^k, \lambda^k, \pi^k)$ à região \mathcal{G} .

Na Figura 5 é representada a direção de viabilidade das restrições de igualdade.

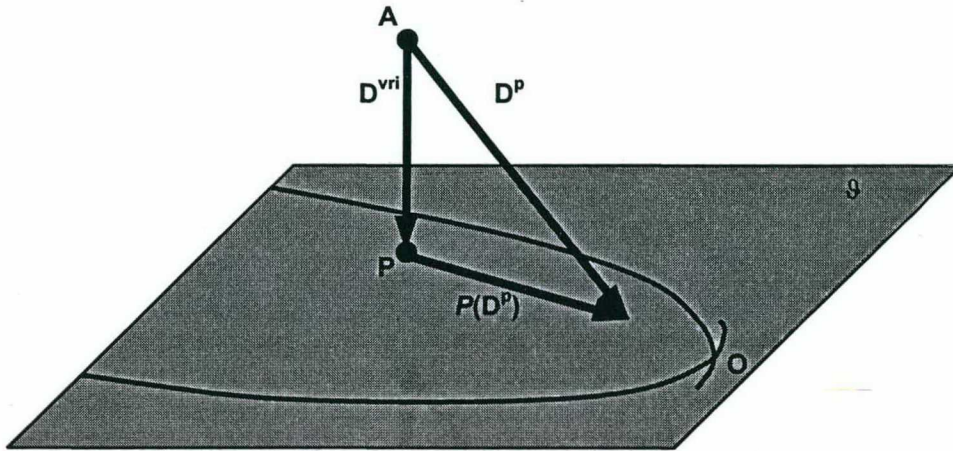


Figura 5: Representação da Direção de Viabilidade das Restrições de Igualdade.

Na Figura 5, a direção $\mathbf{D}^p = \{ \Delta x^k, \Delta s^k, \Delta \lambda^k, \Delta \pi^k \}$, solução das equações (4.6) à (4.8), é a *Direção Perturbada*. Ela visa levar o ponto da iteração atual $\mathbf{A} = (x^k, s^k, \lambda^k, \pi^k)$ ao ponto solução $\mathbf{O} = (x^o, s^o, \lambda^o, \pi^o)$ através de uma direção perturbada pelo parâmetro μ . $\mathbf{D}^{vri} = \{ \Delta x^{vri}, 0, 0, 0 \}$ representa a *Direção de Viabilidade das Restrições de Igualdade*. A direção \mathbf{D}^{vri} é a componente da direção \mathbf{D}^p que visa levar ao ponto \mathbf{A} à região \mathcal{G} , para que na próxima iteração as restrições de igualdade $g(x) = 0$ sejam satisfeitas. A direção $P(\mathbf{D}^p)$ é a projeção da direção \mathbf{D}^p na superfície \mathcal{G} .

Na Figura 5 têm-se também representadas as projeções das restrições de desigualdade $h(x) \leq 0$ na região \mathcal{G} , através de linhas contínuas.

4.4.2. O Caminho Central.

Para um valor constante de $\mu > 0$ no processo iterativo, o algoritmo que resolve as equações perturbadas não-lineares (4.1) à (4.5) tem como solução o ponto $(x(\mu), s(\mu), \lambda(\mu), \pi(\mu))$, distinto da solução candidata ao ótimo $(x^o, s^o, \lambda^o, \pi^o)$. A solução $(x(\mu), s(\mu), \lambda(\mu), \pi(\mu))$ é chamada de *ponto central* e está associada ao valor do parâmetro $\mu \geq 0$. O conjunto dos pontos centrais para distintos valores de μ define o *Caminho Central* ou *Trajétória Central*, como a representada na equação (4.25).

$$l: \mu > 0 \rightarrow (x(\mu), s(\mu), \lambda(\mu), \pi(\mu)) \quad (4.25)$$

O caminho central atravessa o conjunto dos pontos interiores às restrições de desigualdade, mantendo sempre uma distância razoável destas restrições.

Nos algoritmos primais-duais, o caminho central pode ser definido através do valor da média das distâncias primais-duais, como definidas na Seção 4.2.

Com o objetivo de representar graficamente o caminho central, suponham-se as seguintes considerações:

Análise de duas restrições de desigualdade. A fim de limitar o número de desigualdades na representação, são consideradas duas equações $h_1(x) \leq 0$ e $h_2(x) \leq 0$ ativas no ponto solução $(x^o, s^o, \lambda^o, \pi^o)$;

Estudo das variáveis primais. O método de Pontos Interiores visa resolver simultaneamente os problemas primal e dual. Com o objetivo de melhorar a visualização, suponha-se a representação exclusiva do problema primal, com as variáveis duais consideradas como constantes. Uma análise similar pode ser realizada para o problema dual;

Representação das direções na superfície \mathcal{G} . Todas as direções consideradas incluem a componente devida à viabilidade das restrições de igualdade. Entretanto, como o caminho central considera fundamentalmente as restrições de desigualdade, este caminho pode ser analisado através das projeções das direções na superfície \mathcal{G} . Por simplicidade de notação, na representação das curvas nesta superfície não é utilizada a notação de projeção. Assim, a curva $P(h_1)$ é apresentada como h_1 .

Na Figura 6 mostra-se o caminho central para o problema de otimização, com as considerações anteriores.

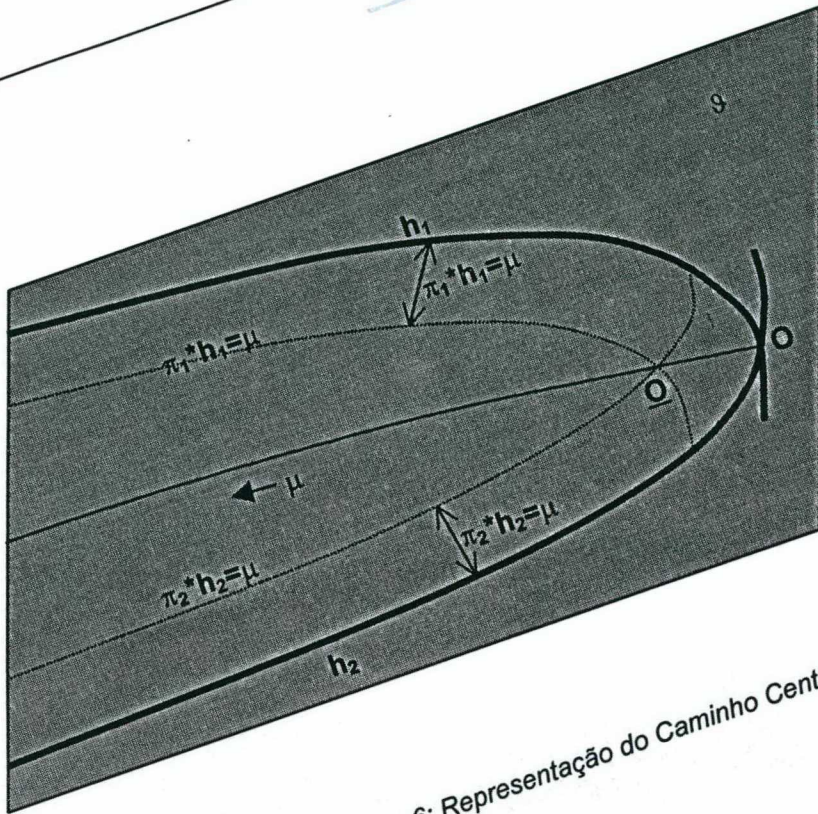


Figura 6: Representação do Caminho Central.

Na Figura 6, h_1 e h_2 representam as projeções no plano 9 dos limites da região de convergência definida pelas restrições de desigualdade do problema considerado ($h_1(x) = 0$ e $h_2(x) = 0$). As restrições são ativas no ponto solução O , ou seja, $h_1(x^0) = 0$ e $h_2(x^0) = 0$, com multiplicadores de Lagrange em geral não nulos. Definindo-se um valor qualquer μ , é possível representar a região dos pontos afastados uma distância dos pontos distando μ das curvas limites são curvas paralelas às restrições, representadas na Figura 6 com linhas tracejadas. A interseção destas curvas para um valor de μ é o ponto \underline{O} . Este ponto representa a solução de um problema (eq. (4.1)-(4.5)) com uma perturbação constante μ . Para distintos valores de μ , são encontrados distintos pontos \underline{O} . Estes pontos determinam uma curva central, conhecida como Caminho Central.

O Caminho Central tem duas interpretações:

- É o lugar geométrico dos pontos solução dos problemas com perturbação μ ,

ar dos pontos afastados uma distancia primal-dual constante μ das restrições no ponto solução.

ção seguinte é analisada a direção de busca do Método de Pontos Interiores, em relação às restrições de desigualdade.

4.3. As Direções Afim-Escala e de Centralização.

Na Figura 6 o Caminho Central é representado para um problema com as mesmas condições supostas na seção 4.4.2, ou seja, duas restrições de desigualdade ativas na solução, projetadas na superfície de viabilidade das restrições de igualdade. Neste mesmo plano, são mostradas as direções Afim-Escala e de Centralização analisadas na Seção 4.3.

Suponha-se que numa iteração i , o ponto $A = \{x^i, s^i, \lambda^i, \pi^i\}$ tenha sido obtido. A Figura 7 apresenta as direções de Centralização, Afim-Escala e de Pontos Interiores convencional no referido ponto.

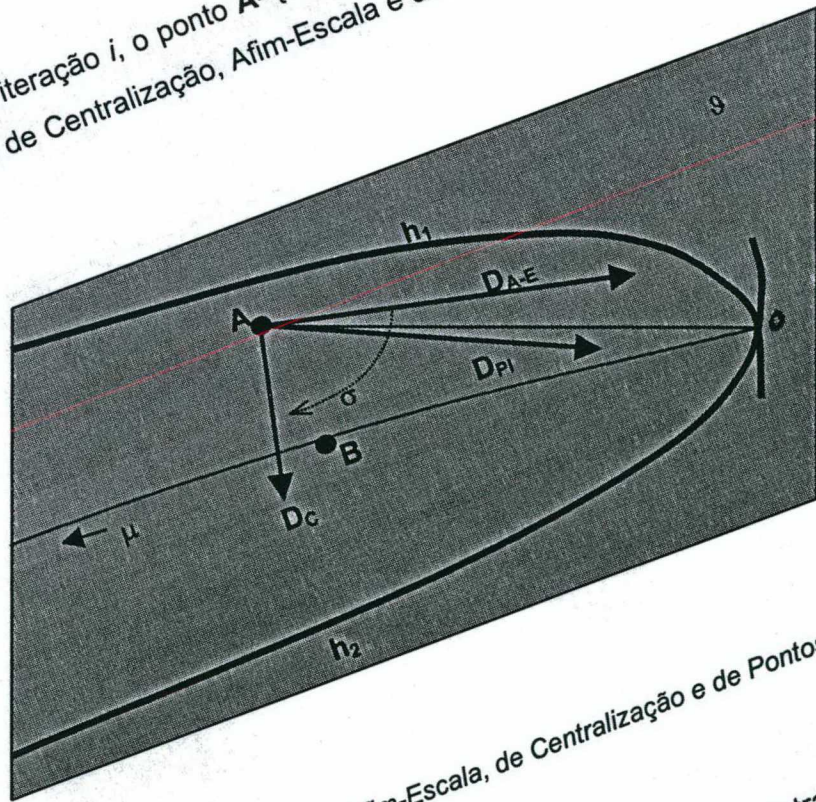


Figura 7: Direções Afim-Escala, de Centralização e de Pontos Interiores.

Na Figura 7, D_c , D_{A-E} e D_{PI} representam as direções de Centralização, Afim-Escala e de Pontos Interiores, respectivamente. Como mostrado na Seção 4.3, a direção Afim-Escala procura atingir o ponto solução O através de uma linearização das equações (4.3.1).

com perturbação μ nula. Devido à linearização, em geral a direção D_{A-E} não coincide com a reta \overline{AO} , direção ótima ao ponto solução. Pelo contrário, a Direção de Centralização D_C resulta da linearização do mesmo sistema de equações com perturbação μ não-nula. Esta direção visa atingir o Caminho Central num ponto B , através da linearização das equações (4.1)-(4.4) expressa pelas equações (4.6)-(4.8). O ponto de solução B depende do valor da perturbação μ na iteração.

Conforme visto na Seção 4.3, o Método de Pontos Interiores realiza em cada iteração uma combinação linear das direções de Centralização e Afim-Escala. O parâmetro de combinação das direções σ determina a percentagem de cada direção que é utilizada na direção de busca do Método de Pontos Interiores. Um valor de σ próximo a zero aproxima a direção de busca do método à direção Afim-Escala e valores de σ próximos a um priorizam a centralização do método com uma percentagem maior da Direção de Centralização (Figura 7). Valores intermediários nesse parâmetro afetam a velocidade de convergência, podendo reduzir o número de iterações requerido para atingir o ponto solução O .

A determinação de um valor ótimo no parâmetro de combinação das direções σ representa uma pesquisa aberta na área da otimização matemática. Alguns dos métodos propostos na literatura para a determinação deste parâmetro são analisados no presente trabalho, estudando-se as suas características e propriedades de convergência. Em geral, estas versões do Método de Pontos Interiores são propostas para problemas de programação linear, requerendo a sua transformação para a aplicação no problema não-linear de FPO. A proposição destes métodos é mostrada nas seções seguintes, e sua aplicação na seção de Resultados.

4.5. Os Algoritmos de Trajetória Central.

4.5.1. Conceitos Básicos.

4.5.1.1. Região de Viabilidade das Restrições de Desigualdade.

A região de Viabilidade das Restrições de Desigualdade é formada pelos pontos interiores às restrições de desigualdade do problema de otimização, conforme expresso pela equação (4.26).

$$\chi = \{x, h(x) \leq 0\} \quad (4.26)$$

Na Figura 8, mostra-se a região de Viabilidade das Restrições de Desigualdade para o problema formulado na Seção 4.4.2.

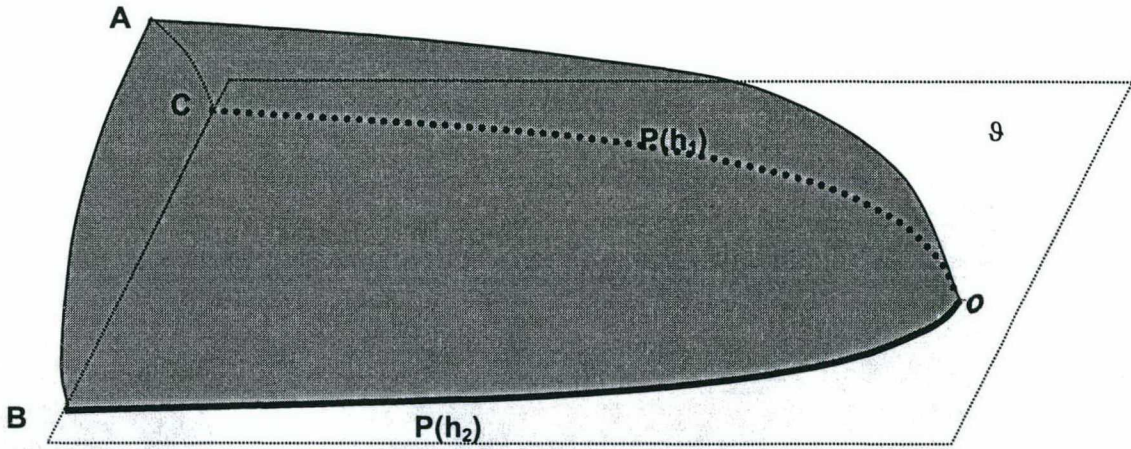


Figura 8: Região de Viabilidade das Restrições de Desigualdade.

Na Figura 8, \mathcal{g} representa o plano de viabilidade das restrições de igualdade, ou seja, a superfície $g(x) = 0$; O é ponto solução do problema de otimização; a superfície ABO é a região dos pontos x nos quais $h_2(x) = 0$; ACO é a superfície na qual $h_1(x) = 0$; a curva BO é a interseção entre o plano de viabilidade das restrições de igualdade e a superfície ABO , ou seja, a projeção $P(h_2(x))$ da restrição de desigualdade $h_2(x)$ no plano \mathcal{g} ; a curva CO é a projeção $P(h_1(x))$ no plano \mathcal{g} ; e a curva AO é a interseção das superfícies $h_1(x) = 0$ e $h_2(x) = 0$.

O volume com fronteira na superfície $ABOC$ é a região dos pontos que satisfazem $h(x) \leq 0$, ou seja, é a região de viabilidade das restrições de desigualdade.

4.5.1.2. Proximidade à Trajetória Central.

Conforme mostrado na Seção 4.4.2, para um especificado $\mu > 0$, a trajetória central deve satisfazer $[\pi]^* s = \mu^* e$ (equação (4.4)). Portanto, numa iteração qualquer é possível

avaliar o desvio desta condição, definindo-se as *proximidades primal-dual à trajetória central* como:

$$\delta(\pi, s, \mu) = \left\| \frac{[\pi]^* s}{\mu} - e \right\| \quad (4.27)$$

$$\delta_{\infty}(\pi, s, \mu) = \left\| \frac{[\pi]^* s}{\mu} - e \right\|_{\infty} \quad (4.28)$$

onde: $\delta(\pi, s, \mu)$ e $\delta_{\infty}(\pi, s, \mu)$ são as proximidades à trajetória central, definidas para a norma euclideana e infinita, respectivamente.

Estas proximidades definem vizinhanças da trajetória central, medidas em múltiplos de μ . Dado $(\alpha, \mu) \geq 0$, as *vizinhanças pequena* e *grande* são definidas respectivamente como,

$$N_{\alpha} = \{(\pi, s), \delta(\pi, s, \mu) \leq \alpha\} \quad (4.29)$$

$$N_{\alpha_{\infty}} = \{(\pi, s), \delta_{\infty}(\pi, s, \mu) \leq \alpha\} \quad (4.30)$$

4.5.1.3. Algoritmo Geral

O algoritmo geral dos métodos de Pontos Interiores Primal-Dual para PNL é apresentado na Seção 3.6 do Capítulo 3. A seguir, é mostrada uma síntese deste algoritmo, utilizada para a formulação das versões do método.

Algoritmo 4- 1: Algoritmo Geral dos Métodos de Pontos Interiores em PNL.

1. Inicialização: Seja um ponto inicial $(x^1, s^1, \lambda^1, \pi^1)$, com $(s^1, \pi^1) > 0$. Seja $\mu^1 > 0$.
2. Montagem e ordenação simbólica do sistema linear.
3. Resolução do sistema linear e as equações auxiliares.
4. Cálculo dos fatores de passo.
5. Atualização das variáveis.

6. Determinação do parâmetro μ .
 7. Teste de Convergência: se convergência: FIM; se não-convergência: CONTINUA.
 8. Atualização do sistema linear.
 9. Retornar ao passo 3.
-

A maioria das versões do método de Pontos Interiores estudados no presente trabalho varia na forma da determinação do parâmetro μ durante o processo iterativo, passo 6 do Algoritmo 4- 1. Além disso, em geral essas versões requerem duas substituições direta-inversa por iteração.

Este trabalho propõe o estudo de alguns algoritmos de trajetória central e sua aplicação ao problema de FPO. A descrição destes algoritmos é apresentada na seção seguinte.

4.5.2. Algoritmo de Direções Sucessivas.

Para cada iteração do algoritmo de Direções Sucessivas (VILLELA, 1997), avalia-se se uma especificada medida de proximidade à trajetória central $\delta < \xi$ é satisfeita. Em caso positivo, é utilizado um passo afim-escala. Caso contrário, é usado um passo de centralização. O caminho de convergência é formado de direções Afim-Escala e de Centralização sucessivas, porém não necessariamente alternadas.

O algoritmo resultante é apresentado a seguir.

Algoritmo 4- 2: Direções Sucessivas.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$. Seja $\mu^l > 0$. Seja $\xi > 0$.
2. Determinação do parâmetro σ .
 - a) Cálculo da proximidade $\delta(x, s, \mu)$.
 - b) Se $(\delta(x, s, \mu) \leq \xi)$ então $\sigma = 1$. Caso contrário, $\sigma = 0$.

3. Cálculo do parâmetro μ com σ determinado no passo 2.
 4. Montagem e ordenação simbólica do sistema linear.
 5. Resolução do sistema linear e as equações auxiliares.
 6. Cálculo dos fatores de passo.
 7. Atualização das variáveis.
 8. Teste de Convergência: Convergência: FIM; Não-convergência: CONTINUA.
 9. Atualização do sistema linear.
 10. Retornar ao passo 2.
-

4.5.2.1. Análise do Algoritmo.

No passo 1 do Algoritmo 4- 2, é requerida a inicialização do parâmetro $\xi > 0$, que determina a proximidade do ponto $(x^k, s^k, \lambda^k, \pi^k)$, na iteração k , ao Caminho Central. Este valor pré-definido, determina a trajetória de convergência do algoritmo. Valores elevados de ξ originam trajetórias predominantemente determinadas pelas direções afim-escala, aproximando o comportamento do algoritmo ao método de Newton. Valores relativamente pequenos no parâmetro ξ exigem direções de busca próximas à trajetória central, com numerosas iterações de centralização.

No passo 2 do mesmo algoritmo é determinada a proximidade de $(x^k, s^k, \lambda^k, \pi^k)$ em relação ao caminho central. Se o ponto se encontra próximo à trajetória central é realizada uma iteração sob a direção Afim-Escala ($\sigma = 1$), tentando aproximar $(x^k, s^k, \lambda^k, \pi^k)$ à solução ótima $(x^o, s^o, \lambda^o, \pi^o)$. Se o ponto está afastado do caminho central é realizada uma iteração de centralização ($\sigma = 0$), visando na próxima iteração uma trajetória melhor ao ponto solução $(x^o, s^o, \lambda^o, \pi^o)$. Como o processo de centralização do ponto $(x^k, s^k, \lambda^k, \pi^k)$ ao caminho central é um processo iterativo, é possível que seja requerida mais de uma iteração de centralização para satisfazer $(\delta(x, s, \mu) \leq \xi)$, realizando o Algoritmo 4- 2 mais de uma iteração de centralização antes de efetuar uma busca na direção Afim-Escala.

No passo 3, é calculado o valor de μ , dependendo da conveniência de se realizar um passo de Centralização ou Afim-Escala.

Os passos seguintes do Algoritmo 4- 2 são iguais aos realizados no algoritmo convencional (Algoritmo 4- 1).

4.5.3. Algoritmo de Direções Sucessivas Fortemente Centralizado.

Em cada iteração do algoritmo de Direções Sucessivas Fortemente Centralizado (VILLELA, 1997) verifica-se se uma tolerância da proximidade é satisfeita, isto é, $\delta < \xi$. Em caso positivo, utiliza-se um passo Afim-Escala. Caso contrário, é realizada uma seqüência de passos de centralização até se atingir $\delta < \zeta < \xi$. Esta trajetória é similar ao algoritmo de Direções Sucessivas, porém realiza uma melhor centralização da trajetória. O algoritmo é mostrado a seguir.

Algoritmo 4- 3: Direções Sucessivas Fortemente Centralizado.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$. Seja $\mu^l > 0$. Seja $(\xi, \zeta) > 0$. Seja $Flag = 1$.
2. Determinação do parâmetro σ .
 - a) Cálculo da proximidade $\delta(x, s, \mu)$.
 - b) Se $(\delta(x, s, \mu) \geq \xi)$ então $Flag = 0, \sigma = 1$.
 Caso contrário,
 Se $(Flag = 1)$ então $\sigma = 0$
 Caso contrário,
 Se $(\delta(x, s, \mu) \geq \zeta)$ então $\sigma = 1$
 Caso contrário, $\sigma = 0, Flag = 1$.
3. Cálculo do parâmetro μ com σ determinado no passo 2.
4. Montagem e ordenação simbólica do sistema linear.
5. Resolução do sistema linear e as equações auxiliares.

6. Cálculo dos fatores de passo.
 7. Atualização das variáveis.
 8. Teste de Convergência: se convergência: FIM; se não-convergência: CONTINUA.
 9. Atualização do sistema linear.
 10. Retornar ao passo 2.
-

4.5.3.1. Análise do Algoritmo.

No passo 1 do Algoritmo 4- 3 é requerida a pré-definição do parâmetro $\zeta > 0$. O parâmetro ξ determina a distância máxima que a trajetória de busca do ponto ótimo $(x^o, s^o, \lambda^o, \pi^o)$ deve se encontrar do caminho central a fim de realizar um passo afim-escala. O parâmetro ζ determina o valor de tolerância no processo iterativo de centralização. A variável *Flag* sinaliza o processo realizado: se *Flag*=1, procura-se o ponto ótimo através de iterações afim-escala. Entretanto, se *Flag*=0, o algoritmo se encontra num processo iterativo de centralização da trajetória até atingir a tolerância ζ .

No passo 2 do mesmo algoritmo, é realizada a determinação do cálculo da proximidade ao Caminho Central. Se uma dada tolerância não é satisfeita, o processo de centralização é iniciado, através da mudança no valor de *Flag*. Quando é atingida a tolerância requerida para a centralização, o procedimento é encerrado, permitindo-se a utilização de passos afim-escala.

4.5.4. Algoritmo de Pontos Interiores Centralizado.

Neste tipo de algoritmo, conforme descrito na referência (VILLELA, 1997), a cada iteração a proximidade $\delta < \xi$ é avaliada. Se a proximidade satisfaz a tolerância, prossegue-se numa direção similar à do método de Pontos Interiores convencional, com $0 < \sigma < 1$. No caso de um afastamento maior do que o permitido pela tolerância, realiza-se um passo de centralização. Uma variante deste algoritmo, a qual pode ser chamada de Pontos Interiores Fortemente Centralizado, realiza uma seqüência de passos de centralização até atingir a trajetória central, enquanto a medida da proximidade não é satisfeita. O algoritmo de Pontos Interiores Centralizado é mostrado a seguir.

Algoritmo 4- 4: Pontos Interiores Centralizado.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$. Seja $\mu^l > 0$. Seja $\xi > 0$. Seja $\sigma_{PI} > 0$
 2. Determinação do parâmetro σ .
 - c) Cálculo da proximidade $\delta(x, s, \mu)$.
 - d) Se $(\delta(x, s, \mu) \leq \xi)$ então $\sigma = \sigma_{PI}$. Caso contrário, $\sigma = 0$.
 3. Cálculo do parâmetro μ com σ determinado no passo 2.
 4. Montagem e ordenação simbólica do sistema linear.
 5. Resolução do sistema linear e as equações auxiliares.
 6. Cálculo dos fatores de passo.
 7. Atualização das variáveis.
 8. Teste de Convergência: se convergência: FIM; se não-convergência: CONTINUA.
 9. Atualização do sistema linear.
 10. Retornar ao passo 2.
-

4.5.4.1. Análise do Algoritmo.

No passo 1 do Algoritmo 4- 4, é requerida a inicialização dos parâmetros $\xi > 0$ (que determina a proximidade ao caminho central) e $\sigma_{PI} > 0$ (composição das direções de centralização e afim-escala utilizada no método de Pontos Interiores convencional).

O algoritmo alterna não sucessivamente as direções de Centralização com uma composição das direções predeterminada. As direções de busca deste algoritmo possuem sempre uma percentagem de centralização, favorecendo em geral suas características de convergência em relação ao Algoritmo 4- 2.

4.5.5. Otimização na Direção de Busca.

SUN e ZHAO (1998) propõem uma alternativa na determinação do máximo passo na direção de busca. Os algoritmos analisados anteriormente visam determinar uma direção de busca que satisfaça um determinado parâmetro, alternando duas direções. Opcionalmente, o algoritmo proposto por SUN e ZHAO (1998) visa obter o ponto ótimo numa direção pré-especificada, segundo um critério de otimização.

No trabalho de SUN e ZHAO (1998) é demonstrada a convergência quadrática desta alternativa para problemas não-lineares de variação monotônica com restrições de desigualdade, utilizando-se a direção do método de Pontos Interiores convencional. Em cada iteração, após a determinação da direção, é obtido o ponto que minimiza as distâncias primais-duais na próxima iteração, atualizando-se as variáveis neste ponto.

Seja $(\Delta x, \Delta s, \Delta \lambda, \Delta \pi)$ a direção de variação numa iteração do método de Pontos Interiores convencional, para um valor de σ pré-especificado. A atualização das variáveis na iteração seguinte é obtida através das expressões (4.31)-(4.34).

$$x = x + \kappa * \alpha_p * \Delta x = x + \Delta x' \quad (4.31)$$

$$s = s + \kappa * \alpha_p * \Delta s = s + \Delta s' \quad (4.32)$$

$$\lambda = \lambda + \kappa * \alpha_d * \Delta \lambda = \lambda + \Delta \lambda' \quad (4.33)$$

$$\pi = \pi + \kappa * \alpha_d * \Delta \pi = \pi + \Delta \pi' \quad (4.34)$$

onde $\kappa = 0.99995$ é o fator de redução dos passos, e $(\Delta x', \Delta s', \Delta \lambda', \Delta \pi')$ é a variação efetiva na direção de busca, considerando-se os passos α_p e α_d .

SUN e ZHAO (1998) propõem a introdução de um parâmetro $0 \leq \theta \leq 1$ nas equações (4.31)-(4.34), de forma a se obter as expressões (4.35)-(4.38).

$$x = x + \theta * \Delta x' \quad (4.35)$$

$$s = s + \theta * \Delta s' \quad (4.36)$$

$$\lambda = \lambda + \theta * \Delta \lambda' \quad (4.37)$$

$$\pi = \pi + \theta * \Delta \pi' \quad (4.38)$$

O valor de θ nas equações (4.35)-(4.38) é obtido de modo a minimizar a distância primal-dual $(\pi^t * s)$ na iteração seguinte. Calculando-se a distância primal-dual através das expressões (4.36) e (4.38), obtém-se (4.39).

$$\pi(\theta)^t * s(\theta) = (\pi + \theta * \Delta\pi^t)^t * (s + \theta * \Delta s) \quad (4.39)$$

A equação (4.39) é uma expressão quadrática de θ , da qual pode se obter o valor de $0 \leq \theta \leq 1$ que minimize (4.39). Para o cálculo do mínimo, é requerido que a função (4.39) seja convexa. Após a determinação de θ , as variáveis são atualizadas segundo equações (4.35)-(4.38).

Apesar da convergência do algoritmo mostrada em SUN e ZHAO (1998) para a direção do método de Pontos Interiores convencional, a busca linear na direção pode ser aplicada para qualquer outra das variantes estudadas. O algoritmo geral é formulado a seguir.

Algoritmo 4- 5: Pesquisa na Direção de Busca.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$.
 2. Obtenção da direção de busca do método.
 3. Cálculo dos fatores de passo.
 4. Determinação do parâmetro θ pela resolução da equação (4.39).
 5. Atualização das variáveis.
 6. Determinação do parâmetro μ .
 7. Teste de Convergência: se convergência: FIM; se não-convergência: CONTINUA.
 8. Atualização do sistema linear.
 9. Retornar ao passo 2.
-

4.5.5.1. Análise do Algoritmo.

Após a obtenção dos fatores de passo, realiza-se no Algoritmo 4- 5 uma otimização na direção de busca a fim de minimizar a distância primal-dual média na próxima iteração. Para isto, é acrescentado o passo 4 antes da atualização das variáveis.

4.5.6. Algoritmo Preditor-Corretor de Mehrotra.

Desde a sua formulação por MEHROTRA (1992), o algoritmo Preditor-Corretor vem sendo estudado como uma alternativa ao algoritmo de Pontos Interiores Primal-Dual convencional para PNL.

A cada iteração do algoritmo Preditor-Corretor Primal Dual de Pontos Interiores para PNL (WU *et al.*, 1994, WEI *et al.*, 1998) duas etapas são realizadas.

Etapas de predição: consiste na determinação das distâncias primais-duais na próxima iteração especificando $\sigma = 0$. A etapa preditora corresponde à solução de um problema não-perturbado ($\mu = 0$). Nesta etapa, são resolvidas as equações (4.6)-(4.8) e os fatores de passo correspondentes(eq. (3-27) e (3-28) do Capítulo 3), calculando-se os resultados da predição (\underline{s}^{as} , $\underline{\lambda}_g^{as}$) através das expressões (4.40)-(4.41).

$$s^{as} = s^{as} + \kappa * \alpha_p * \Delta s \quad (4.40)$$

$$\pi^{as} = \pi^{as} + \kappa * \alpha_d * \Delta \pi \quad (4.41)$$

Etapas de correção: nesta etapa, duas modificações são realizadas:

- a) Expansão das equações de complementaridade (4.4) até o termo de segunda ordem da série de Taylor, segundo a equação:

$$[s] * \Delta \pi + [\pi] * \Delta s + [\Delta s] * \Delta \pi = -[\pi] * s + \mu * e \quad (4.42)$$

Desde que o produto $[\Delta s] * \Delta \pi$ é não-linear, o algoritmo Preditor-Corretor usa a

informação da fase de predição para estimar este termo. Então, a equação (4.42) é modificada para

$$[s] * \Delta\pi + [\pi] * \Delta s = -[\pi] * s + \mu * e - [\Delta s^{as}] * \Delta\pi^{as} \quad (4.43)$$

As modificações expressas pela equação (4.43) são refletidas nas expressões (4.8) e (4.10), sendo substituídas pelas equações (4.44) e (4.45), respectivamente.

$$\Delta\pi = -\pi + \left[\frac{I}{s} \right] * \mu * e - \left[\frac{\pi}{s} \right] * \Delta s - \left[\frac{\Delta s^{as}}{s} \right] * \Delta\pi^{as} \quad (4.44)$$

$$t_i(x, s, \lambda, \pi) = \nabla L(x, s, \lambda, \pi) + \nabla h(x)' * \left\{ \left[\frac{\pi}{s} \right] * h(x) + \left[\frac{I}{s} \right] * \mu * e - \left[\frac{\Delta s^{as}}{s} \right] * \Delta\pi^{as} \right\} \quad (4.45)$$

b) Determinação dinâmica do parâmetro de combinação das direções em cada iteração através da expressão (4.46).

$$\sigma = \left(\frac{(\pi^{as})' * s^{as}}{(\pi)' * s} \right)^3 \quad (4.46)$$

Na equação (4.46), os produtos $\{(\pi)' * s\}$ e $\{(\pi^{as})' * s^{as}\}$ representam as distâncias primais-duais na iteração atual e na direção de predição, respectivamente. Esta expressão é usada para se estimar dinamicamente a necessidade da centralização na trajetória para a solução ótima. Se na predição $\{(\pi^{as})' * s^{as}\} \approx \{(\pi)' * s\}$, significa que o avanço na direção Afim-Escala não consegue melhorar a média das distâncias primais do ponto atual. Por conseguinte, prefere-se utilizar na iteração uma direção próxima à direção de Centralização (eq. (4.47)).

$$\sigma = \left(\frac{(\pi^{as})' * s^{as}}{(\pi)' * s} \right)^3 \approx 1 \quad (4.47)$$

Por outro lado, se o avanço na direção Afim-Escala resulta numa redução significativa das distâncias primais-duais, esta direção é preferencialmente utilizada, de acordo com a expressão (4.48).

$$\sigma = \left(\frac{(\pi^{as})' * s^{as}}{(\pi)' * s} \right)^3 \approx 0 \quad (4.48)$$

O algoritmo Preditor-Corretor do Método de Pontos Interiores para PNL apresentado anteriormente é formulado a seguir.

Algoritmo 4- 6: Preditor-Corretor.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$.
2. Montagem do sistema linear (4.6)-(4.8) com $\sigma = 0$.
3. Etapa de Predição:
 - a. Resolução do sistema linear.
 - b. Cálculo dos passos (eq. (3-27) e (3-28) do Capítulo 3).
 - c. Armazenamento dos resultados da predição nas variáveis auxiliares (s^{as}, π^{as}) , equações (4.40)-(4.41).
4. Etapa de Correção.
 - a. Cálculo de σ : eq. (4.46).
 - b. Cálculo de μ : eq. (4.12).
 - c. Expansão de segunda ordem nas equações de complementaridade: Modificação do lado direito do sistema linear através das equações (4.44) e(4.45).
 - d. Resolução do sistema linear.

- e. Atualização das variáveis
 - f. Cálculo dos passos (eq. (3-27) e (3-28) do Capítulo 3).
5. Teste de Convergência: se convergência: FIM; se não-convergência: CONTINUA.
 6. Atualização do sistema linear com $\sigma = 0$.
 7. Retornar ao passo 3.
-

4.5.6.1. Análise do Algoritmo.

No passo 1 do Algoritmo 4- 6, observa-se que apenas um ponto inicial com variáveis de folga e multiplicadores de Lagrange das restrições de desigualdade positivos é requerido. O parâmetro σ , definido pelo usuário no algoritmo convencional, é determinado dinamicamente pelo método em cada iteração, dependendo da vantagem da utilização da direção Afim-Escala.

No passo 3, são calculadas as variáveis (s^{as}, π^{as}) através da direção afim-escala e no passo 4 é calculado o novo ponto da trajetória. O sistema linear a ser resolvido no passo 4 é diferente daquele utilizado no passo 3, pela aproximação de segunda ordem das equações de complementaridade e a modificação do valor de μ . Entretanto, estas alterações modificam somente o lado direito do sistema linear, preservando a matriz de coeficientes já fatorada. Os incrementos nas variáveis no passo 4 são assim obtidas numa nova substituição direta-inversa.

Em resumo, o algoritmo Preditor-Corretor procura fornecer uma melhor estimativa para a combinação das direções de Centralização e Afim-Escala. Para isto, é requerido o conhecimento do valor das distâncias primais-duais via direção afim-escala, o que demanda duas substituições direta-inversa do sistema linear em cada iteração. Este esforço computacional adicional torna atrativa a utilização de ferramentas de Alto Desempenho na sua resolução.

4.5.7. Algoritmo Preditor-Corretor com Múltiplas Correções.

A representação por equações de segundo grau nas equações de complementaridade permite uma melhor aproximação ao comportamento não-linear destas expressões. No

trabalho de SHANNO e VANDERBEI (1999) é utilizada mais de uma etapa de correção por iteração. O número de correções a serem realizadas pode ser pré-estipulada pelo usuário ou estimado segundo um determinado critério. Na versão aqui apresentada, o número de etapas corretoras por iteração é fixo e determinado por duas condições: o número máximo de correções permitidas e o efetivo decréscimo da distância primal-dual com a adição da correção. O procedimento é formulado a seguir em termos algorítmicos.

Algoritmo 4- 7: Preditor-Corretor, Múltiplas Correções.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$. Seja $n_corr > 0$.
2. Montagem do sistema linear (4.6)-(4.8) com $\sigma = 0$.
3. Etapa de Predição:
 - a. Resolução do sistema linear.
 - b. Cálculo dos passos (eq. (3-27) e (3-28) do Capítulo 3).
 - c. Armazenamento dos resultados da predição nas variáveis auxiliares (s^{as}, π^{as}) , equações (4.40)-(4.41). Armazenamento da distância media primal-dual na variável auxiliar $dmpd_a$.
4. Faz $corr = 0$; $Flag = 1$.
5. Etapa de Correção. Se $((corr < n_corr) \text{ and } (Flag = 1))$, faz
 - a. $corr = corr + 1$.
 - b. Cálculo de σ : eq. (4.46).
 - c. Cálculo de μ : eq. (4.12).
 - d. Expansão de segunda ordem nas equações de complementaridade: Modificação do lado direito do sistema linear através das equações (4.44) e(4.45).
 - e. Resolução do sistema linear.
 - f. Atualização das variáveis
 - g. Cálculo dos passos (eq. (3-27) e (3-28) do Capítulo 3).
 - h. Cálculo da distância primal-dual $dmpd$ no novo ponto.
 - i. Se $(dmpd > dmpd_a)$, faz $Flag = 0$.
 - j. $dmpd_a = dmpd$

6. Teste de Convergência: se convergência: FIM; se não-convergência: CONTINUA.
 7. Atualização do sistema linear com $\sigma = 0$.
 8. Retornar ao passo 3.
-

4.5.7.1. Análise do Algoritmo.

No passo 1 do Algoritmo 4- 7, é necessária a pré-especificação (pelo usuário) do número de máximo de etapas de correção permitidas por iteração. No passo 5, são realizadas n_corr etapas de correção, as quais aproximam os valores $(\Delta s^{as}, \Delta \pi^{as})$ na equação (4.43) aos resultados exatos $(\Delta s, \Delta \pi)$, recalculando os valores de σ .

Em cada iteração são requeridas $(n_corr + 1)$ substituições direta-inversa, n_corr devidas às etapas de correção mais uma na etapa de predição (passos 4 e 5 do algoritmo). Entretanto, somente uma fatoração é requerida em cada iteração.

4.5.8. Algoritmo de Máximo Passo na Trajetória Central

O método de Máximo Passo na Trajetória Central visa atingir o ponto solução através de uma trajetória próxima ao caminho central. GONZAGA (1997) propõe a determinação da distância à trajetória central através das medidas das distâncias primais-duais no ponto, como expressa a eq. (4.49).

$$\delta(\pi, s, \mu) = \left\| \frac{\pi^t * s}{\mu} - e \right\| \quad (4.49)$$

O algoritmo de GONZAGA (1997) é formulado para o método de Pontos Interiores Linear, realizando-se no presente trabalho uma avaliação de seu comportamento na programação não-linear do FPO.

Partindo de um ponto inicial qualquer, o algoritmo visa obter pontos da trajetória a uma distância ε pré-especificada pelo usuário. Assim, na iteração $(k+1)$, deve ser satisfeita a igualdade (4.50).

$$\delta(\pi^{k+1}, s^{k+1}, \mu^{k+1}) = \left\| \frac{(\pi^{k+1})^t * s^{k+1}}{\sigma * dmpd} - e \right\| = \varepsilon \quad (4.50)$$

O método visa encontrar valores da variável σ que combinem as direções afim-escala e de Centralização a fim de satisfazer a igualdade (4.50). Os valores $(x^{k+1}, s^{k+1}, \lambda^{k+1}, \pi^{k+1})$ na próxima iteração dependem dos passos α_p e α_d a serem calculados posteriormente. Portanto, na equação (4.50), o ponto da trajetória na próxima iteração pode ser aproximado pela equação (4.51).

$$(x^{k+1}, s^{k+1}, \lambda^{k+1}, \pi^{k+1}) \approx (x, s, \lambda, \pi) + (\Delta x, \Delta s, \Delta \lambda, \Delta \pi) \quad (4.51)$$

De acordo com a equação (4.51), o produto das distâncias primais-duais na próxima iteração pode ser calculado através da expressão (4.52).

$$(\pi^{k+1})^t * s^{k+1} = \pi^t * s + \pi^t * \Delta s + \Delta \pi^t * s + \Delta \pi^t * \Delta s \quad (4.52)$$

A substituição da equação (4.8) na (4.52) resulta na expressão (4.53).

$$(\pi^{k+1})^t * s^{k+1} = \mu * e + \Delta \pi^t * \Delta s \quad (4.53)$$

Substituindo (4.53) em (4.50), resulta (4.54).

$$\delta(\pi^{k+1}, s^{k+1}, \mu^{k+1}) = \left\| \frac{\Delta \pi^t * \Delta s}{\sigma * dmpd} \right\| = \varepsilon \quad (4.54)$$

As direções de modificação nas variáveis são obtidas através da combinação das direções afim-escala e de Centralização (eq. (4.21) e (4.22)), as quais substituídas na expressão (4.54) resultam na equação (4.55).

$$\delta(\pi^{k+1}, s^{k+1}, \mu^{k+1}) = \left\| \frac{(\sigma * \Delta\pi^c + (1-\sigma) * \Delta\pi^{ae})' * (\sigma * \Delta s^c + (1-\sigma) * \Delta s^{ae})}{\sigma * dmpd} \right\| = \varepsilon \quad (4.55)$$

Para um valor pré-determinado de ε a única variável da equação (4.55) é σ , o parâmetro de combinação das direções.

Utilizando-se a norma euclidiana na equação (4.55), a obtenção do valor de σ reduz-se à solução de uma equação de quarta ordem. Como possuem significado físico apenas os valores reais de σ no intervalo (0,1), as raízes correspondentes podem ser obtidas através do método de Bisseção (BARBEAU, 1989, GONZAGA, 1997).

Quando utilizada a norma infinita na equação (4.55), é requerida a resolução de uma série de equações de segundo grau, tantas quantas forem as restrições de desigualdade consideradas no problema. Das raízes reais obtidas é escolhida a maior em magnitude no intervalo (0,1).

O algoritmo é mostrado a seguir.

Algoritmo 4- 8: Máximo Passo na Trajetória Central.

1. Inicialização: Seja um ponto inicial $(x^l, s^l, \lambda^l, \pi^l)$, com $(s^l, \pi^l) > 0$. Seja $\varepsilon > 0$.
2. Direção Afim-Escala: Montagem e resolução do sistema linear (4.6)-(4.8) com $\sigma = 0$. Armazenamento do resultado nas variáveis $(\Delta x^{as}, \Delta s^{as}, \Delta \lambda^{as}, \Delta \pi^{as})$.
3. Direção de Centralização: modificação do lado direito e resolução do sistema linear (4.6)-(4.8) com $\sigma = 1$. Armazenamento do resultado nas variáveis $(\Delta x^c, \Delta s^c, \Delta \lambda^c, \Delta \pi^c)$.
4. Obtenção de σ : resolução da eq. (4.55).
5. Combinação das direções:

$$(\Delta x, \Delta s, \Delta \lambda, \Delta \pi) = \sigma * (\Delta x^c, \Delta s^c, \Delta \lambda^c, \Delta \pi^c) + (1-\sigma) * (\Delta x^{as}, \Delta s^{as}, \Delta \lambda^{as}, \Delta \pi^{as})$$
6. Atualização das variáveis.

7. Teste de Convergência: se convergência: FIM; se não-convergência: CONTINUA.
 8. Retornar ao passo 2.
-

4.5.8.1. Análise do Algoritmo.

No passo 1 do Algoritmo 4- 8, é requerida a pré-especificação pelo usuário do valor de $\varepsilon > 0$. O valor de proximidade máxima à trajetória central ε durante o processo iterativo é constante e modifica a trajetória percorrida até o ponto ótimo. Um parâmetro de combinação das direções, σ , que satisfaça esta condição, é determinado automaticamente no algoritmo.

Em cada iteração do algoritmo devem ser calculadas as direções de centralização e afim-escala no ponto corrente, requerendo uma substituição direta-inversa adicional. Por conseguinte, os tempos de cálculo computacional por iteração são superiores em geral àqueles obtidos com o método de pontos interiores convencional.

As melhores características de centralização da trajetória podem resultar atrativas em problemas com regiões de convergência estreitas, devendo-se considerar as características particulares do problema.

4.6. Conclusões.

Este capítulo analisa a direção de busca do método de Pontos Interiores, decompondo-a em três direções, em função das restrições do problema. A participação de duas destas direções, a de Centralização e a Afim-Escala, é controlada pelo Parâmetro de Combinação das Direções. Distintos critérios de combinação podem ser utilizados, constituindo as diferentes versões do método. Estes distintos algoritmos permitem variar algumas das características de convergência da versão convencional.

No capítulo seguinte é considerada a vetorização do algoritmo. Esta vetorização é ainda mais importante para as novas versões, por requerer substituições direta-inversa adicionais em cada iteração do processo iterativo.

Capítulo 5.

Vetorização do Método de Pontos Interiores

5.1. Introdução

As aplicações do FPO podem requerer elevados tempos de cálculo, fazendo conveniente uma redução nos tempos de processamento a fim de viabilizar a sua utilização na operação em tempo real. Para esta finalidade, propõe-se o uso de Processamento de Alto Desempenho, metodologia que permite dividir o esforço computacional requerido nas operações.

Conforme observado anteriormente, o Método de Pontos Interiores possui boas características em termos de tratamento das restrições de desigualdade e da convergência. Entretanto, cada iteração é computacionalmente onerosa, devido à necessidade de armazenar e resolver um sistema linear de ordem superior a quatro vezes o número de barras do sistema. O esforço computacional resultante é uma das principais dificuldades para a aplicação do método na operação de sistemas de potência em tempo real. Duas possibilidades são previstas para melhorar os tempos computacionais: aprimorar o método de Pontos Interiores no sentido de reduzir o número de iterações (como apresentado no Capítulo 4) e utilizar o Processamento de Alto Desempenho (PAD).

5.2. Processamento de Alto Desempenho.

Desde a sua proposição na década de 60, as técnicas de PAD paralela e vetorial vêm sendo utilizadas para melhorar o desempenho dos métodos, possibilitando soluções factíveis em termos de tempos computacionais. A partir dos anos 70, a comunidade científica começou a avaliar o potencial dos computadores paralelos e vetoriais. As primeiras máquinas foram poucas em número, e disponíveis somente para alguns

laboratórios de instituições estrangeiras. Na atualidade, a crescente expansão dessas máquinas, e no Brasil a possibilidade de uso da Rede Nacional de Pesquisa, permitem a utilização de PAD para os trabalhos das distintas áreas de conhecimento. O custo dos processadores vetoriais e paralelos tem diminuído razoavelmente com a aplicação de novas tecnologias, de forma que estes processadores começam a ser usados na resolução de problemas com alto custo computacional (QUINN, 1987).

Tendo-se disponível os processamentos vetorial e paralelo, a decisão sobre qual deles utilizar depende basicamente do problema a ser resolvido. A paralelização é preferível quando tem-se um problema de grande granularidade, passível de divisão em blocos quase independentes, com pouca ou nenhuma comunicação. Por outro lado, a vetorização é preferível quando é realizado um grande número de operações vetor-vetor, em laços de grande comprimento. Ambas as técnicas requerem uma reavaliação dos procedimentos utilizados nos processamentos escalar ou seqüencial.

A formação e a solução eficiente de um sistema linear de grande dimensão a cada iteração constitui uma das principais dificuldades para a aplicação do método de Pontos Interiores no problema de FPO. Particularmente, a montagem e atualização deste sistema é uma tarefa de alto custo computacional, pois envolve numerosas operações em ponto flutuante, e a resolução de um grande número de operações aritméticas entre matrizes e vetores esparsos. Em geral, a solução de sistemas lineares não possuem uma alta granularidade o que dificulta a utilização eficiente de técnicas de processamento paralelo. Isto é particularmente certo nos métodos de Pontos Interiores de PNL, que utilizam no seu processo iterativo matrizes com alto número de condicionamento. Uma das principais características do processamento vetorial é a sua capacidade de propiciar "speed-ups" sem interferir na convergência do método, mantendo invariante o número de iterações para a convergência. Neste trabalho, ele melhora a rapidez do processamento computacional nos setores mais críticos do processo iterativo: a solução e atualização do sistema linear.

5.3. Principais Características do Processamento Vetorial

O processamento vetorial baseia-se no conceito de "pipelining", ou seja, na capacidade de realizar tarefas similares simultâneas ao mesmo tempo (QUINN, 1987). Uma vez preparados os vetores envolvidos numa operação, o tempo de realização de uma operação simples entre eles é quase um intervalo de tempo de resolução, fornecendo tantos resultados quanto for o comprimento dos vetores envolvidos. Tal como a idéia de

paralelização, é um conceito simples, mas muitas vezes difícil de se concretizar eficientemente na prática.

Os computadores vetoriais utilizam o conceito de “pipelining” através de unidades funcionais, dedicadas cada uma a um propósito específico. As operações vetorizáveis pelo computador consistem de instruções simples incluídas em laços tipo DO.

5.3.1. Operações Vetorizáveis.

O tipo de operações passíveis de serem vetorizadas dependem das unidades funcionais presentes no computador. Em geral, são realizadas somas, subtrações, multiplicações, divisões, e outras operações entre dois vetores. Os laços tipo DO são utilizados em Fortran para explicitar operações vetoriais, pelo que são utilizados na detecção de operações que podem ser vetorizadas. Entretanto, existem condições que impedem a vetorização das operações, ainda presente a correspondente unidade funcional. Algumas delas são:

- Existência de outro laço interior;
- Chamada a algum procedimento de Entrada-Saída de dados de alguma unidade externa (impressão, leitura de dados de um arquivo, etc.);
- Instruções condicionais (IF, etc.);
- Utilização de variáveis caracteres;
- Diretrizes para expressamente inibir a vetorização (CDIR@ NEXTSCALAR, etc.);
- Direcionamentos do interior do laço a instruções no exterior (GOTO);
- Recorrências numéricas que produzam diferentes resultados em modo escalar do modo vetorial;
- Referências ambíguas aos índices de laço, ou variações muito complexas para calcular em tempo de execução;
- Chamada a função de biblioteca que não possui uma versão vetorial.

A aparição de algumas das condições anteriores inibe a vetorização automática do laço correspondente. Entretanto, uma adequada modificação na estrutura da operação pode permitir a vetorização através de procedimentos específicos. Algumas das condições

anteriores são consideradas posteriormente na vetorização dedicada através do “caminho” de fatoração. Um estudo mais profundo de ditas regras, e de condições especiais onde o hardware do computador permite vetorizar, encontra-se na referência (CRAY, 1994).

5.3.2. O Tempo de Preparação da Operação.

Suponha-se uma operação simples vetorizável entre dois vetores **A** e **B** de comprimento n numa unidade funcional com capacidade de armazenamento de m segmentos. No primeiro intervalo de tempo, um par de valores (um por cada vetor) é fornecido à unidade funcional, sendo armazenados nas primeiras posições dos vetores de armazenamento interno da unidade. Nos intervalos seguintes, pares de valores são fornecidos à unidade, até preencher a capacidade de armazenamento de m pares. No intervalo de tempo $m+1$, o primeiro resultado é fornecido pela unidade. O tempo requerido até que o primeiro valor é fornecido pela unidade é conhecido como Tempo de Preparação, dependendo o seu valor das operações necessárias para a determinação de que a operação é vetorizável, da capacidade de armazenamento e dos tempos de entrada e saída dos dados na unidade, entre outros fatores. Numa forma simplificada, se $n \gg m$ existirá ganho na utilização da unidade funcional, pois após o tempo de preparação será fornecido um resultado da operação por intervalo de tempo, sendo os procedimentos de leitura e escrita dos dados realizados simultaneamente com o cálculo. Se $n \leq m$, os tempos da operação vetorizada resultarão superiores à implementação escalar (tradicional) da operação. A dimensão de armazenamento m requerida pela unidade depende dos tempos de leitura e escrita dos dados na memória.

Observa-se da análise anterior que a dimensão dos vetores envolvidos numa operação é determinante na decisão da vetorização de uma operação. Um maior comprimento dos vetores permite a diminuição da importância dos tempos de preparação, permitindo ganhos superiores. Indexações nos vetores que redundem em elevados tempos de leitura e/ou escrita dos dados podem também prejudicar a performance da vetorização.

5.3.3. Encadeamento das Unidades Funcionais.

Os computadores vetoriais contem várias unidades funcionais. Dependendo de sua complexidade, as unidades funcionais possuem diferente quantidade de segmentos. Como o número de segmentos determina o número de períodos até o primeiro resultado,

as operações mais complexas, que possuem maior quantidade de segmentos, tem menor ganho vetorial que as mais simples.

Outra característica interessante dos processadores vetoriais é a sua capacidade de encadear operações vetoriais. Suponha-se que requeira-se numa linha de instrução a soma de dois vetores **A** e **B** e a multiplicação elemento a elemento do resultado da soma por um terceiro vetor **C**, todos os vetores de dimensão n . Esta operação é mostrada no Algoritmo 5-1.

Algoritmo 5-1: Soma e Multiplicação de Vetores.

1. DO $i = 1, n$
 2. $D(i) = (A(i) + B(i)) * C(i)$
 3. END DO
-

No Algoritmo 5-1, o vetor **D** armazena o resultado da operação. Suponha-se que as unidades funcionais de soma e multiplicação possuam tempos de preparação dos vetores t_1 e t_2 , respectivamente. Como anteriormente explicado, estes tempos são requeridos para que o primeiro resultado seja fornecido pela unidade funcional, e dependem do número de segmentos de cada unidade. Uma vez fornecido o primeiro par de valores dos vetores **A** e **B** à unidade de soma, o processador deve aguardar t_1 intervalos de tempo até que o primeiro resultado da soma seja disponível. Este primeiro resultado é fornecido à unidade de multiplicação com o primeiro valor do vetor **C**. Assim, o primeiro resultado da operação completa é fornecido em $(t_1 + t_2)$ intervalos de tempo. Após este tempo de preparação da operação, um resultado é fornecido por intervalo. Isto representa um ganho substancial na operação de vetores de grande comprimento, pois numerosas leituras e escritas de dados e várias operações algébricas são realizadas simultaneamente em cada intervalo.

Existem 7 unidades independentes de operações vetoriais no CRAY Y-MP, mas não se recomenda nas instruções de uso a utilização de mais de dois encadeamentos, por problema de procura e armazenamento de dados na memória (CRAY, 1994).

5.3.4. Transferência dos Dados

Os tempos envolvidos na transferência de dados podem, em certas situações, até superar os tempos próprios do cálculo. Isto pode acontecer na utilização de indexações indiretas, em dois ou três níveis. Todos os computadores após o Cray I contém um hardware especial que permite a utilização de indexação indireta, porém, os custos de busca e armazenamento aconselham não utilizar dimensionamentos indiretos desnecessários. Neste trabalho, procura-se a diminuição das indexações indiretas do algoritmo.

Outro fator importante que afeta a eficiência do algoritmo vetorial são as locações dos dados envolvidos no cômputo. É importante que os dados afetados sejam contínuos, ou próximos. Para uma análise aprofundada deste aspecto da vetorização, recomenda-se CRAY (1994) e LEWIS (1993).

5.3.5. A Vetorização Automática

Em resumo, uma operação simples entre dois vetores **A** e **B** é realizada vetorialmente no computador nos seguintes passos esquemáticos:

- Identificação da existência da operação vetorizável;
- Determinação das unidades lógicas envolvidas;
- Hierarquização das unidades lógicas na operação;
- Fornecimento de um dado por vetor á unidade lógica de menor hierarquização por intervalo de tempo;
- Após o tempo de preparação da operação, leitura de um resultado por intervalo de tempo.

Estas operações constituem a Vetorização Automática realizada pelo computador. Quando as condições de vetorização de uma operação são satisfeitas (Seção 5.3.1) o computador realiza os passos anteriores a fim de vetorizar a operação, sempre que esta não seja expressamente desabilitada.

A fim de melhorar a Vetorização Automática do computador, pode-se notar que é necessária uma adaptação dos algoritmos a vetorizar, antes e durante a programação,

para adequá-los aos procedimentos que são mais eficientemente realizados nos processadores vetoriais.

Operações complexas de serem vetorizadas requerem procedimentos específicos, como os vistos na seção seguinte.

5.4. Vetorização de Operações Complexas

As operações simples, como aquelas mostradas no Algoritmo 5-1, podem ser vetorizadas a nível de instrução. Operações mais complicadas, como a solução de sistemas lineares, requerem procedimentos específicos com metodologias particulares para sua vetorização. Os procedimentos de fatoração e substituição direta-inversa, requeridos na solução de sistemas lineares, necessitam principalmente de operações do tipo mostrado na equação (5.1).

$$a_i \leftarrow b_i * a_i \quad (5.1)$$

Na equação (5.1), é lida uma posição de memória, modificado o seu valor pela multiplicação de outra variável, e o resultado novamente armazenado na posição de memória. A operação representada em tal equação pode ser vetorizada, através das unidades funcionais do computador. A eficiência da vetorização depende do número de operações (5.1) incluídas no laço. Segundo DONGARRA *et al.* (1998), os algoritmos de vetorização da solução de sistemas lineares podem ser divididos em:

Algoritmos de Vetorização Frontal

Uma vez escolhido pelo processo de ordenação um pivô, na fatoração de um sistema linear recalculam-se os elementos da matriz alocados em linhas e/ou colunas de ordem posterior no processo de ordenação. Este recálculo depende do valor original do elemento, e dos valores na linha e coluna do pivô. Assim, este procedimento pode ser expresso na forma da equação (5.1), num laço denominado Frente. Um procedimento similar acontece com a substituições direta e inversa. Os algoritmos que utilizam este desacoplamento no cálculo para a vetorização dos procedimentos de solução de sistemas lineares são denominados de Algoritmos de Vetorização Frontal.

FORREST e TOMLIN (1990) utilizam um algoritmo frontal para a vetorização do método Simplex e uma versão do método de Pontos Interiores Linear. Os testes são realizados num computador IBM 3090 e se mostraram mais favoráveis com matrizes densas. A fim de melhorar a performance destes algoritmos vetoriais, uma outra família de métodos de vetorização é utilizada.

Algoritmos de Vetorização Multi-Frontal

Os métodos que tentam incluir mais de um pivô no mesmo laço vetorizável são denominados de Vetorização Multi-Frontal. Estes algoritmos utilizam alguns critérios para encontrar independência no cálculo dos elementos em diferentes frentes, dependendo das características da matriz utilizada. Duas metodologias principais têm sido usadas para a vetorização da solução de sistemas lineares na área de Sistemas de Potência: o método da Matriz-W (HUANG e LU, 1993), na qual a relação entre as operações é reduzida pelo uso de fatores inversos; e a estratégia que utiliza o "caminho" de fatoração (MONTAGNA, GRANELLI e VUONG, 1996; MONTAGNA *et al.*, 1992), procedimento utilizado neste trabalho.

5.5. Utilização do Processamento Vetorial em Sistemas de Potência

Os métodos da Matriz-W (PADILLA e MORELATO, 1992; HUANG e LU, 1993) e do "caminho" de fatoração (MONTAGNA, GRANELLI e VUONG, 1996; MONTAGNA *et al.*, 1992), têm sido aplicados à vetorização da resolução do Jacobiano do Fluxo de Potência convencional.

Em 1990, ENNS, TINNEY e ALVARADO propõem a solução do fluxo de potência através do método da matriz W. HUANG e LU (1993) utilizam este método para a vetorização da solução do sistema linear de um fluxo de potência. Testes são realizados em computadores Convex 3840 e Allinat FX/80, e em sistemas de grande porte (1400 barras) são obtidos "speed-ups" de 3.3. GOMEZ e BETANCOURT (1990) implementam um Fluxo de Potência Desacoplado Rápido com vetorização via matriz W, com testes num computador Cray X/MP e "speed-ups" entre 3 e 4 para o total do processo iterativo.

MONTAGNA *et al.*, (1992), mostram a aplicação do método de "caminho" de fatoração à vetorização do fluxo de potência convencional em computadores IBM 3090 e Cray Y/MP. Testes foram realizados com o sistema IEEE 118 barras e com distintas configurações

reais, com "speed-ups" entre 5 e 8 para os procedimentos vetorizados. Num trabalho posterior, MONTAGNA, GRANELLI e VUONG (1996), utilizando computadores Cray Y-MP e X-MP2, testam sistemas reais de grande porte (quase 12.000 barras) atingindo ganhos de até 15 vezes na fatoração e 10 vezes nas substituições.

Entretanto, o problema proposto da vetorização do FPO possui características que o diferenciam da vetorização do fluxo de potência convencional. Entre outras particularidades, a matriz a ser resolvida neste último é geralmente diagonal dominante, com pivôs não-nulos em todas as linhas/colunas da matriz, o que não acontece com a matriz do FPO via método de Pontos Interiores. A matriz aqui utilizada é formada da união de diferentes blocos diagonal-dominantes e possui um bloco nulo no canto inferior, como mostrado na Figura 4 do Capítulo 3.

A adaptação vetorial de um procedimento originalmente implementado para o cálculo escalar, requer modificações numa série de etapas do algoritmo. É necessário reprogramar o método visando à obtenção de laços possíveis de vetorizar e, em algumas áreas do programa essencialmente escalares, obter procedimentos dedicados ao cálculo vetorial. Parte deste trabalho é uma extensão da pesquisa desenvolvida na dissertação de mestrado (CASTRONUOVO, 1997), na qual foram obtidos os seguintes resultados:

- Implementação usando a linguagem FORTRAN de um algoritmo de Pontos Interiores Primal-Dual convencional, versão PNL, considerando esparsidade e esquemas de ordenação;
- Utilização do conceito de "caminho" de fatoração para a vetorização do procedimento de substituição direta-inversa na resolução de sistemas lineares;
- Avaliação do algoritmo em sistemas teste reduzidos: IEEE 6, 14, 30 e 57 barras.

A formulação do Método de Pontos Interiores convencional utilizada na referência CASTRONUOVO, (1997) é diferente daquela apresentada no Capítulo 3. Naquele trabalho, formulou-se o algoritmo de solução com base na adição de uma função barreira na função Lagrangeana, como mostrado em (GRANVILLE, 1984; SANCHEZ, 1983). Através de uma função barreira logarítmica, penalizava-se a vizinhança aos limites das

restrições de desigualdade e, após manipulações matemáticas, eram obtidas expressões similares às apresentadas no Capítulo 3.

Devido à pequena dimensão dos sistemas lineares envolvidos, os resultados da vetorização da substituição direta-inversa não se mostraram muito significativos. Os ganhos obtidos em tempo computacional para esta rotina estavam na ordem de 1.5, comparando-se a vetorização automática do computador e a vetorização proposta. Tendo-se realizado somente a vetorização das substituições, os "speed-ups" para o total da convergência eram reduzidos.

Os resultados anteriores levaram a propor a vetorização da fatoração do sistema linear (rotina de alto custo computacional) e a aplicação do algoritmo em sistemas reais de médio e grande porte. Além disso, uma re-vetorização total do algoritmo foi realizada a fim de reduzir os tempos computacionais envolvidos.

5.6. Procedimentos de Vetorização

Dependendo do tipo de operação a ser vetorizada, diversas técnicas foram utilizadas na vetorização do FPO via método de Pontos Interiores. Muitas destas vetorizações consistiam na reordenação de operações para permitir satisfazer as condições de vetorização apresentadas na Seção 5.3.1. Isto reduz os tempos computacionais requeridos para a solução do problema de otimização. Entretanto, as operações automaticamente vetorizadas não fornecem ganhos ao algoritmo, porque a comparação é realizada com essa versão.

Algumas vetorizações específicas realizadas no algoritmo são mostradas a seguir.

Cálculo e armazenamento dos coeficientes do sistema linear.

As operações matemáticas que se repetem numerosamente na mesma ordem podem ser vetorizadas pelo histórico. Após a primeira realização da operação essa ordem é conhecida, podendo ser armazenada para uma vetorização. Como exemplo, considere-se a vetorização entre duas matrizes esparsas A e B e o armazenamento do resultado na matriz C , procedimento realizado numerosas vezes na determinação dos coeficientes do sistema linear. Na primeira vez que esta operação é realizada, utiliza-se o procedimento tradicional de cálculo, que inclui a busca dos elementos a serem multiplicados nas estruturas esparsas das matrizes e a determinação da posição na qual é armazenado o resultado na matriz C . Estes valores das posições nas listas encadeadas são

armazenados na ordem da operação, fornecendo um histórico das buscas dos elementos para a resolução da multiplicação. Na vez seguinte que é requerida a multiplicação entre as matrizes esparsas **A** e **B**, ou entre duas outras com a mesma estrutura, basta percorrer o histórico para realizar a operação, eliminando-se a necessidade das buscas nas estruturas esparsas das matrizes e diminuindo-se o tempo de processamento. Este tipo de vetorização requer três vetores auxiliares inteiros de dimensões relativamente grande, pelo que, apesar de ser muito eficiente na redução dos tempos de cálculo, deve ser utilizado cuidadosamente. Neste trabalho, a vetorização da multiplicação de matrizes esparsas através do histórico é utilizada na formação da submatriz Hessiana, na qual são realizadas quatro multiplicações de matrizes esparsas com a mesma estrutura, por iteração.

Cálculo das equações auxiliares.

As equações auxiliares são calculadas após a resolução do sistema linear reduzido, para a determinação das variações nas variáveis de folga e os multiplicadores de Lagrange das restrições de desigualdade. A vetorização do cálculo dos multiplicadores de Lagrange (eq. (3-22)) é direta e realizada automaticamente pelo computador, por consistir na soma e multiplicação de vetores. A vetorização da determinação dos incrementos nas variáveis de folga (eq. (3-21)) é mais complicada, e depende se a restrição é simples ou funcional. Como anteriormente explicado, restrições simples de desigualdade representam limites nas variáveis de otimização, e restrições funcionais são os limites em funções destas variáveis.

A equação (3-21) ($\Delta s = -(h(x) + s) - \nabla h(x) * \Delta x$) expressa o cálculo na variação das variáveis de folga como a subtração de dois vetores, um deles calculado através da multiplicação das derivadas das restrições de desigualdade vezes os incrementos nas variáveis primais de otimização. Numa restrição simples i , a linha $\nabla h_i(x)$ possui um só elemento não-nulo, permitindo a vetorização direta e automática da operação. Sendo $\nabla h(x)$ armazenada em forma esparsa, a existência de mais de um elemento na linha (como acontece com as restrições de desigualdade funcionais) requer de operações de busca dentro da coluna. Devido a que no problema estudado as restrições funcionais aparecem em menor número que as restrições de desigualdade simples, só é realizada neste trabalho a vetorização das últimas.

Vetorização da Resolução do Sistema Linear Reduzido

Nesta vetorização é utilizado o conceito de "caminho" de fatoração a fim de diminuir os tempos computacionais na operação. Esta técnica é analisada na seção seguinte.

5.7. Vetorização da Solução do Sistema Linear

5.7.1. Introdução.

Algumas das características que dificultam a vetorização do sistema linear do FPO numa vetorização frontal são a existência de pivôs nulos e os poucos elementos por linha/coluna, que diminuem os ganhos numa vetorização frontal. Entretanto, a configuração da matriz do sistema linear permitem tirar proveito da aplicação do processamento vetorial numa forma multi-frontal, devendo ser enfatizado que a elevada esparsidade e a disposição predominantemente em linhas oblíquas resulta numa conectividade baixa, o que implicitamente indica "caminhos" de fatoração pouco profundos, e de grande número de elementos por nível. Numa implementação eficiente, o comprimento dos vetores envolvidos deverá ser grande, possibilitando assim redução nos tempos de cálculo comparados à implementação escalar. Esta propriedade é explorada no processo de solução do sistema linear via "caminho" de fatoração.

O procedimento de vetorização via "caminho" de fatoração explora a independência no cálculo dos elementos num mesmo nível do grafo de fatoração. Dado que a ordem de substituição para os elementos num mesmo nível é indistinta para o resultado final, todas as operações numéricas de um nível do "caminho" de fatoração são agrupadas num mesmo laço vetorizável, sempre que não ocorram recorrências numéricas de processamento. As possíveis recorrências, que fornecem resultados inexatos na vetorização, são eliminadas através de vetores auxiliares. No presente trabalho é proposta a aplicação do método de "caminho" de fatoração a alguns dos algoritmos apresentados no Capítulo 4, em especial ao Preditor-Corretor. Este algoritmo requer duas substituições direta-inversa em cada iteração, com um esforço computacional elevado.

A matriz do sistema linear do método de Pontos Interiores possui um bloco nulo no canto inferior, ocasionando pivôs nulos. A fim de evitar divisões por zero, é necessário modificar a ordenação tradicional, como apresentada em GOMEZ e FRANQUELO (1988).

Os procedimentos de fatoração e substituições direta/inversa são realizados através de Bi-Fatoração, segundo ZOLLENKOPF (1987).

5.7.2. Vetorização Através dos Níveis do Caminho de Fatoração

Os ganhos em procedimentos vetoriais são obtidos com operações simples executadas em laços com grande número de elementos (DONGARRA *et al.*, 1998). Esta condição dificilmente acontece quando é utilizado armazenamento esparsos. Por conseguinte, a vetorização direta dos algoritmos convencionais não resulta em ganhos apreciáveis. Neste trabalho, a dimensão dos vetores envolvidos nos processos de fatoração LDL^t e substituição direta-inversa do sistema linear é incrementada através do conceito de Nível do "Caminho" de Fatoração (NCF). A substituição direta é usada aqui como um exemplo de vetorização por NCF. O algoritmo convencional para este procedimento é mostrado no Algoritmo 5- 2 (o vetor x é inicializado com os valores do vetor do lado direito do sistema linear).

Algoritmo 5- 2: Substituição Direta Escalar.

```
1. DO I = 1, N
2.     M = ICOL (I)
3.     DO WHILE (M.GT.0)
4.         X(IROW(I)) = X(IROW(I)) - VAL(M) * X(I)
5.         M = INEXT(M)
6.     END DO
7. END DO
```

No Algoritmo 5- 2, a matriz L é armazenada em listas encadeadas, através de colunas, nos vetores seguintes: $ICOL$ (posição do primeiro elemento de cada coluna), $INEXT$ (posição do próximo elemento na coluna), $IROW$ (número de linha do elemento), VAL (valores numéricos dos elementos não-diagonais), e $DIAG$ (elementos diagonais). Os escalares M e I são inteiros auxiliares.

Este algoritmo consiste em dois laços: um laço interno que age com os elementos de uma coluna e um laço externo que opera com as colunas do sistema linear. Na forma mostrada pelo algoritmo, ambos são não-vetorizáveis. O laço que inicia em 1 contém um laço interno, o que não permite a sua vetorização. O laço que começa em 3 tem um comprimento variável, e é portanto não-vetorizável. Neste trabalho, é proposta a vetorização do laço interno através de vetores auxiliares e a inclusão da maioria dos elementos das colunas de um NCF neste laço. Assim, o laço externo é reduzido ao número de níveis do grafo de caminho de fatorização, incrementando-se o número de elementos no laço vetorizável.

Vetorização dos elementos de uma coluna:

O número de elementos de cada coluna pode ser conhecido com antecedência. Considerando que cada elemento é usado só uma vez no procedimento algébrico, é possível vetorizar o laço interno do algoritmo usando ponteiros num algoritmo de Vetorização Frontal. Suponha-se, por exemplo, uma matriz L esparsa fatorizada (Figura 9.a), com um grafo de "caminho" de fatorização mostrado na Figura 9.b.

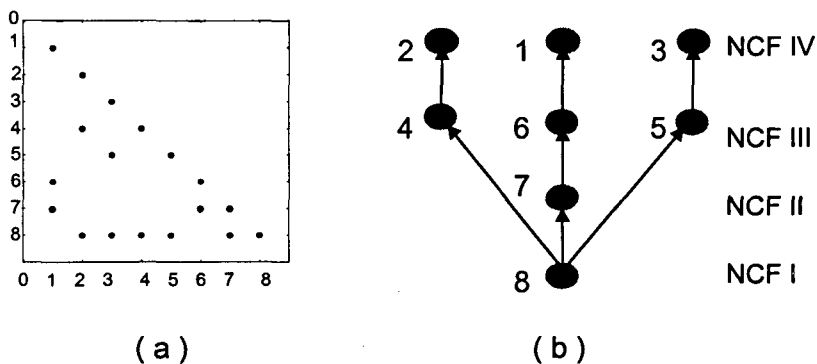


Figura 9: Matriz Exemplo.

A vetorização dos elementos da coluna requer dois vetores auxiliares: IVA (seqüência dos endereços dos elementos na lista encadeada, coluna após coluna) e ILO (primeiro elemento de cada coluna no vetor IVA). Figura 10 mostra estes vetores para a matriz da Figura 9.

ILO= [1,	3,	5,	7,	...]
Elem= [6-1,	7-1,	4-2,	8-2,	5-3,	8-3, ..]
IVA= [21,	4,	9,	3,	6,	11, ..]

Figura 10: Armazenamento dos Elementos da Matriz Exemplo.

Para permitir a vetorização dos elementos de uma coluna, o laço interno do algoritmo de substituição direta é modificado, resultando no algoritmo seguinte.

Algoritmo 5- 3: Substituição Direta, com Vetorização Frontal por Coluna

```

1. DO I = 1, N
2.     DO K = ILO(I), ILO(I+1) - 1
3.         X(IROW(I)) = X(IROW(I)) - VAL(IVA(K)) * X(I)
4.     END DO
5. END DO

```

Utilização do conceito de NCF:

O número de elementos em cada coluna da matriz de coeficientes é pequeno. Assim, o procedimento vetorial é modificado para incluir todas as colunas de um NCF no laço interno. A ordem de eliminação das colunas em um NCF pode ser modificada sem alterar o resultado da operação. Esta independência entre as colunas de um nível pode ser explorada, incluindo todos os elementos das colunas de um NCF num mesmo laço vetorial. Entretanto, esta modificação geralmente resulta num laço recursivo. A leitura ou escrita mais de uma vez de uma posição de memória em um laço vetorial resulta em erros de armazenamento, e deve ser evitada. No NCF IV da matriz da Figura 9.a, esta recorrência acontece com o elemento 8-3, como pode ser observado na Figura 11.

Elem= [6-1,	7-1,	4-2,	8-2,	5-3,	8-3]
IROW= [6,	7,	4,	8,	5,	8]
	↓	↓	↓	↓	↓	☒
IROWa= [6,	7,	4,	8,	5,	erro]

Figura 11: Laço Recursivo no NCF IV.

Dois elementos no laço vetorial pertencentes à mesma linha resultam numa dependência temporária de acesso à memória, com erros na operação, como indicado através de vetor *IROWa* na Figura 11. Neste trabalho, uma modificação dos vetores utilizados na vetorização por coluna permite a eliminação da recorrência numérica no laço vetorial. Os elementos das colunas no mesmo NCF são incluídos, de acordo com sua correspondência, em dois laços: um laço vetorial, com aqueles elementos em linhas diferentes, e um laço escalar contendo os elementos restantes. Os elementos dos dois laços de cada nível são representados no vetor *IVA* em seqüência. Os inícios de cada um destes laços são armazenados no vetor *ILO*. Para a matriz exemplo, a Figura 12 mostra os laços vetorial (*VL*) e escalar (*SL*) do NCF IV.

	VL	SL	
ILO= [1,	6,]

	VL	SL	
Elem= [6-1, 7-1, 4-2, 8-2, 5-3,	8-3]
IVA= [21, 4, 9, 3, 6,	11]

Figura 12: Laços Não Recorrentes do NCF IV.

Os vetores auxiliares são calculados depois da fatorização simbólica e permanecem inalterados nas iterações. O uso de ponteiros permite a aplicação de um procedimento vetorial eficiente na substituição direta, sem reordenar a matriz de coeficientes e com baixo custo computacional. Num procedimento similar ao aplicado à substituição direta, foram vetorizados os procedimentos de fatoração e substituição inversa.

5.8. O Procedimento de Ordenação

A matriz implementada para o FPO neste trabalho possui dimensão $4n-1$, com um bloco nulo no canto inferior, de ordem $2n-ng$, onde n é o número de barras do sistema e ng é o número de geradores. É interessante notar a concentração dos elementos não nulos em linhas oblíquas paralelas à diagonal principal, devido à união de blocos diagonal-dominantes.

Com o objetivo de explorar esta estrutura, o armazenamento compacto através de blocos não é usado (GRANVILLE, 1994, WEI *et al.* 1998, DOMMEL e TINNEY, 1968). Ao invés, listas encadeadas que consideram cada elemento independentemente são aplicadas. Este tipo de armazenamento reduz o número de operações aritméticas de ponto flutuante, porque só os elementos não-nulos são considerados. No armazenamento por blocos, a inversão explícita dos blocos requer em geral de operações com elementos nulos.

A estratégia consiste numa modificação do esquema de ordenação de Mínimo Grau (Tinney 2) (GOMEZ e FRANQUELO, 1988). Inicia-se a ordenação com as colunas com elemento diagonal não-nulo. Posteriormente, são acrescentadas ao procedimento de ordenação as colunas que originalmente possuíam elementos diagonais nulos, e que foram modificados na fatoração por elementos diagonal-dominantes do Jacobiano. Esta metodologia permite considerar as mudanças numéricas no valor dos pivôs durante o processo de fatoração, preservar a estrutura simétrica do sistema linear e a manipulação direta de cada elemento da matriz, requisito necessário para a vetorização proposta.

Os elementos diagonal-dominantes de J são na prática não-nulos durante o processo iterativo, e existe ao menos um deles em cada coluna/linha da matriz. Assim, durante o processo de eliminação destes elementos, os pivôs do bloco nulo modificam o seu valor tornando-se não-nulos. Durante a fatoração simbólica do submatriz J , seus elementos diagonal-dominantes são assinalados por um vetor auxiliar $DOM_J(i)$ com valor 1, com praticamente nenhum custo computacional adicional. Este vetor facilita a identificação destes elementos durante a modificação dos pivôs originalmente nulos.

Na ordenação da matriz segundo o critério de Grau Mínimo (GOMEZ e FRANQUELO, 1988), os seguintes vetores são utilizados: $F(i)$, posição do nó i na ordenação final. Se $F(i) = 0$ significa que o nó ainda não foi considerado na ordenação; $D(i)$, grau do nó i durante o processo de ordenação. Inicialmente, $D(i)$ é o número de elementos não-diagonais na coluna i . Adicionalmente, os seguintes vetores são requeridos a fim de considerar dinamicamente a condição não-nula do pivô: $COND_DIAG(i)$, indica a condição do pivô da coluna i . Se $COND_DIAG(i) = 1$ o pivô é não-nulo e se $COND_DIAG(i) = 0$, significa que a coluna possui originalmente um pivô nulo, e ainda o seu valor não foi modificado por elementos diagonal-dominantes da submatriz J ; $DOM_J(i)$, indica que o elemento é diagonal-dominante na submatriz J .

O algoritmo de Mínimo Grau, modificado a fim de considerar dinamicamente a Condição Não-Nula do Pivô, é mostrado no Algoritmo 5- 4. Em negrito, são mostradas as modificações ao algoritmo apresentado em GOMEZ e FRANQUELO (1988).

Algoritmo 5- 4: Algoritmo de Ordenação de Mínimo Grau, considerando Dinamicamente a Condição Não-nula do Pivô.

1. Inicialização: seja $k = 1$; $F(i) = 0$ para todos os nós; **$COND_DIAG(i) = 0$ para os nós com pivô nulo, em outro caso, $COND_DIAG(i) = 1$** ; $D(i)$, grau do nó.
 2. Encontrar o nó i com $F(i) = 0$ e **$COND_DIAG(i) = 1$** , tal que $D(i)$ é mínimo. Fazer $F(i) = k$.
 3. Por cada nó J adjacente a i tal que $F(j) = 0$, fazer:
 - a. $D(j) = D(j) - 1$.
 - b. **Se o elemento m alocado em (i,j) possui $DOM_J(m) = 1$ e $COND_DIAG(j) = 0$, então fazer $COND_DIAG(j) = 1$.**
 4. Por cada par de nós (m, n) adjacentes a i mas não adjacentes entre eles, tais que $F(m) = F(n) = 0$, adicionar um "fill-in" na posição (m, n) e fazer $D(m) = D(m) + 1$ e $D(n) = D(n) + 1$.
 5. Se $k = N$ então STOP. Outro caso, $k = k + 1$ e voltar ao passo 2.
-

As modificações introduzidas no algoritmo convencional de Mínimo Grau possuem quase nulo esforço computacional e permitem a consideração dinâmica na ordenação do valor

da diagonal. Qualquer ordem na disposição dos pivôs nulos da matriz pode ser considerada, e o método pode ser ampliado para outros esquemas de ordenação, como os apresentados em (GOMEZ e FRANQUELO, 1988).

Deve ser assinalado que, devido à estrutura da matriz que agrupa todos os pivôs nulos em seu canto inferior, a ordenação natural não apresenta problemas de pivô. Entretanto, esta ordenação resulta num número elevado de enchimentos na matriz, com alta conectividade entre as colunas.

5.9. Conclusões

Neste capítulo é apresentada uma metodologia de vetorização que visa reduzir os tempos computacionais do FPO via método de Pontos Interiores. É proposta a modificação do programa computacional a fim de adaptar as instruções a àquelas que podem ser vetorizadas automaticamente pelo computador. Na solução do sistema linear, procedimento de alto custo computacional, é aplicado o conceito de "caminho" de fatoração a fim de implementar um algoritmo de vetorização Multi-Frontal.

Os resultados da aplicação desta metodologia nos algoritmos de Pontos Interiores são mostrados no Capítulo 6.

Capítulo 6.

Resultados

6.1. Introdução

Neste capítulo, as versões do método de Pontos Interiores apresentadas no Capítulo 4 e a vetorização da resolução do sistema linear formulada no Capítulo 5, ambas aplicadas ao problema de FPO, são analisadas. Para isto, o FPO é formulado como um problema de minimização do custo de geração de potência ativa. As redes elétricas utilizadas para a avaliação destas aplicações incluem sistemas teste e realísticos de médio e grande porte, e os resultados são comparados com a versão convencional do método de Pontos Interiores.

6.2. O Fluxo de Potência Ótimo

As metodologias analisadas nos Capítulos 4 e 5 são aplicadas a um problema de FPO não-linear, cuja formulação é representada nas equações (6.1)-(6.6), escritas a seguir.

$$\text{Min. } \sum_{i=1}^{ng} (a_i + b_i * Pg_i + c_i * Pg_i^2) \quad (6.1)$$

$$\text{s.a. } Q^{calc}(V, \theta) + Qd = 0 \quad (6.2)$$

$$Pg - Pd - P^{calc}(V, \theta) = 0 \quad (6.3)$$

$$V^{min} \leq V \leq V^{max} \quad (6.4)$$

$$Pg^{min} \leq Pg \leq Pg^{max} \quad (6.5)$$

$$Qg^{min} \leq (Qd + Q^{calc}(V, \theta)) \leq Qg^{max} \quad (6.6)$$

onde: Pg é o vetor das ng potências ativas de geração, de elementos Pg_i ; (a_i, b_i, c_i) são os coeficientes constante, linear e quadrático das curvas de preço de potência ativa, respetivamente; $Q^{calc}(V, \theta)$ é o vetor das n injeções de potência reativa, de elementos $Q_i^{calc} = V_i * \sum_{j \in K} V_j * (G_{ij} * \text{sen}(\theta_i - \theta_j) - B_{ij} * \text{cos}(\theta_i - \theta_j))$; Pd e Qd são os vetores das n demandas de carga nas barras; $P^{calc}(V, \theta)$ é o vetor das n injeções de potência ativa, de elementos $P_i^{calc} = V_i * \sum_{j \in K} V_j * (G_{ij} * \text{cos}(\theta_i - \theta_j) + B_{ij} * \text{sen}(\theta_i - \theta_j))$; (Pg^{min}, Pg^{max}) são os vetores dos ng limites máximo e mínimo de geração de potência ativa, respetivamente; (Qg^{min}, Qg^{max}) são os vetores dos ng limites máximo e mínimo de geração de potência reativa, respetivamente; (V^{min}, V^{max}) são os vetores dos n limites máximo e mínimo na operação dos módulos de tensão V , respetivamente; θ é o vetor dos n ângulos das tensões nas barras; n e ng é o número de barras e geradores do sistema, respetivamente.

A função objetivo (6.1) visa minimizar os custos de operação do sistema, através da potência ativa requerida. Os preços de venda da potência ativa pelos geradores são representados por curvas quadráticas.

As equações de igualdade (6.2) e (6.3) representam os balanços de potência reativa ($(n-ng)$ equações) e ativa (n equações) no sistema, e correspondem às equações do Fluxo de Potência convencional.

A desigualdade (6.4) expressa os limites no módulo de tensão nas barras requeridos na operação do sistema.

As restrições (6.5) e (6.6) expressam os limites máximo e mínimo na geração de potência ativa e reativa nos geradores do sistema. As ng equações (6.6), $(Qd + Q^{calc}(V, \theta))$ representa em forma implícita a geração de potência reativa nas usinas do sistema.

6.3. Características dos Sistemas Teste

As redes utilizadas neste trabalho são sistemas teste IEEE e reais representando equivalentes da região Sul-Sudeste Brasileira (SSB). As principais características destes sistemas são mostradas na Tabela 1.

Tabela 1: Características dos Sistemas Testes.

Sistema	Número de Linhas	Número de Geradores	Ordem do Sistema Linear
IEEE 30 barras	41	6	119
IEEE 57 barras	80	7	227
IEEE 118 barras	179	34	471
SSB 176 barras	196	30	703
SSB 352 barras	385	30	1407
SSB 750 barras	952	87	2999
SSB 1704 barras	2122	115	6815

A ordem do sistema linear mostrada na Tabela 1 corresponde à matriz reduzida da eq. (3.23) do Capítulo 3. Este sistema linear deve ser resolvido em cada iteração do método de Pontos Interiores. As equações auxiliares, para determinação dos incrementos nas variáveis de folga e seus correspondentes multiplicadores de Lagrange, devem ser resolvidas posteriormente à obtenção da solução do sistema linear.

6.4. Esparsidade dos Sistemas Lineares

A metodologia proposta na Seção 5.7 para o tratamento elemento a elemento dos componentes da matriz de coeficientes do sistema linear é analisada nesta seção. Na Tabela 2 é mostrada a esparsidade inicial e final da matriz, utilizando-se o procedimento de ordenação de Mínimo Grau (Tinney II) modificado a fim de considerar dinamicamente o valor do pivô. Alguns destes resultados são extraídos de CASTRONUOVO *et al.* (1998b).

Tabela 2: Esparsidade

Sistema	Esparsidade Inicial (%)	Esparsidade Final (%)
IEEE 30 barras	90.65	50.67
IEEE 57 barras	94.79	58.88
IEEE 118 barras	97.00	78.17
SSB 176 barras	98.65	92.17
SSB 352 barras	99.32	96.10
SSB 750 barras	99.64	95.61
SSB 1704 barras	99.84	97.78

Nos sistemas teste IEEE mostrados na Tabela 2, se observa a existência de um número elevado de "enchimentos" após a fatoração, os quais reduzem o grau de esparsidade da matriz. Entretanto, nos sistemas reais (de maior porte e mais esparsos do que os sistemas teste) observa-se a eficiência do esquema de ordenação proposto. A esparsidade para estes sistemas é reduzida somente em valores entre 2 e 6% após a fatoração, com valores melhores para os sistemas maiores.

Deve-se salientar que, além do enchimento reduzido, o método possui duas vantagens: 1) é possível manipular elemento a elemento da matriz, requisito da metodologia proposta de vetorização; e 2) é permitida a utilização dos algoritmos convencionais de ordenação com mínimas modificações, sem o uso de blocos.

6.5. O Método de Pontos Interiores Convencional

6.5.1. Resultados de Convergência e Índices de Desempenho

Nesta seção, o método de Pontos Interiores convencional é aplicado ao problema de FPO modelado na Seção 6.2. Também são apresentados alguns dos índices de desempenho utilizados para comparar a convergência dos métodos estudados.

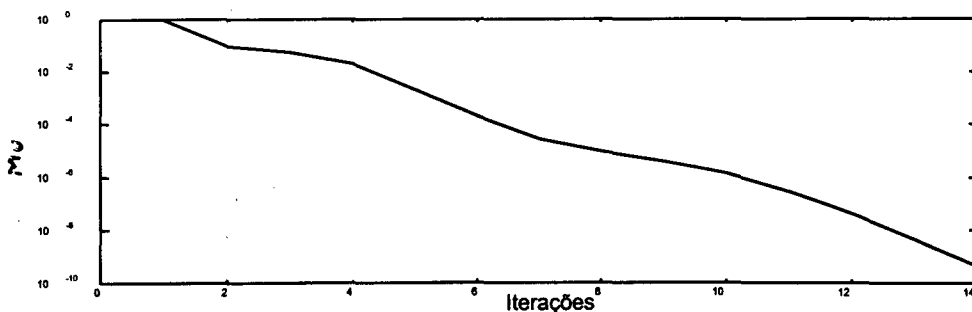
Na Tabela 3 é mostrado o número de iterações até a convergência para a versão convencional do Método de Pontos Interiores, com tolerância de 1×10^{-5} para os

componentes do vetor gradiente da Função Lagrangeana e de 1×10^{-6} para parâmetro μ . Salvo indicação em contrário, os testes para todos os algoritmos foram realizados adotando-se as mesmas tolerâncias.

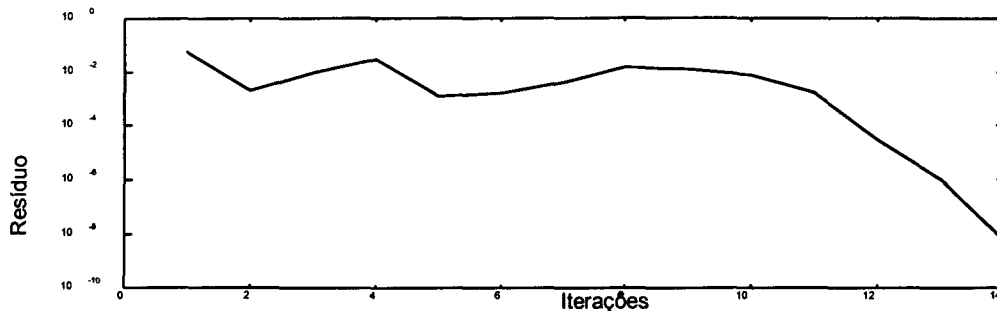
Tabela 3: Algoritmo Convencional - Iterações até a Convergência.

Sistema	Número de Iterações
IEEE 30 barras	10
IEEE 57 barras	11
IEEE 118 barras	14
SSB 176 barras	10
SSB 352 barras	14
SSB 750 barras	25
SSB 1704 barras	21

A Tabela 3 mostra a *não-escalabilidade* no número de iterações do algoritmo de Pontos Interiores. A quantidade de iterações até a convergência não é função da dimensão do sistema, mas depende da complexidade do problema, das características da região de convergência e do ponto inicial. Como exemplo, o sistema linear da rede SSB 1704 é 2.3 vezes maior que o da rede SSB 750, porém requer um número de iterações 16% menor. Na Figura 13 representa-se o valor do parâmetro de perturbação μ e o máximo resíduo nas iterações, para o sistema SSB 352 barras.



(a)



(b)

Figura 13: Algoritmo Convencional - Variação do μ e Máximo Resíduo. SSB 352

Observa-se na Figura 13.a, que o parâmetro de perturbação decresce quase linearmente numa escala logarítmica, diminuindo aproximadamente uma ordem de grandeza por iteração. Por outro lado, o máximo resíduo (Figura 13.b) não apresenta este tipo de variação. O valor do resíduo máximo flutua nas primeiras iterações, decrescendo rapidamente no final do processo iterativo, o que pode ser atribuído às características de convergência dos métodos de Newton na vizinhança da solução ótima.

Um índice de desempenho importante do processo de convergência do algoritmo convencional de Pontos Interiores é a distância entre a trajetória de busca e o Caminho Central, iteração a iteração. Na equação de complementaridade das condições de otimalidade perturbadas de primeira ordem de KKT, para cada desigualdade i a condição $\pi_i * s_i = \mu$ deve ser satisfeita. A distância entre a trajetória de busca e o Caminho Central mede o desvio desta condição. Desde que o parâmetro de perturbação é calculado em função da média das distâncias primais-duais na iteração, algumas das versões analisadas neste trabalho baseiam-se no estabelecimento de um valor máximo para este desvio.

As distâncias ao caminho central em cada iteração para o sistema SSB 350, são apresentadas na Figura 14 (CASTRONUOVO *et al.*, 2000b). Essas distâncias são calculadas conforme mostrado na Seção 4.5.1.2. do Capítulo 4, através da norma Euclidiana.

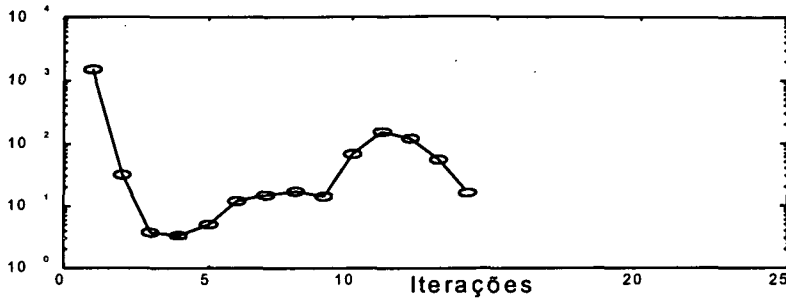


Figura 14: Algoritmo Convencional - Distâncias ao Caminho Central, em unidades de μ . SSB 352 (escala semilogarítmica).

Observa-se na Figura 14, que o método de Pontos Interiores na versão convencional, não mantém uma distância constante ao Caminho Central. Na realidade, em algumas iterações a solução corrente se afasta de forma considerável da trajetória central. Conforme visto anteriormente, esta versão não considera a distância ao Caminho Central na sua formulação, utilizando um parâmetro de combinação das direções empírico e constante durante as iterações.

Os resultados do método de Pontos Interiores convencional são usados neste capítulo para comparação com as outras versões do algoritmo de Pontos Interiores.

6.6. Vetorização do Algoritmo Convencional

6.6.1. Considerações Iniciais.

O número de iterações do método de Pontos Interiores não é afetado grandemente pelo aumento nas dimensões do sistema (Tabela 3). Entretanto, o tempo computacional requerido por iteração cresce juntamente com a dimensão do sistema ou, mais precisamente, com o número de elementos não-nulos da matriz de coeficientes do sistema linear. A resolução de problemas de otimização em sistemas de grande porte pode demandar demasiado esforço computacional para a operação em tempo real, justificando a utilização de ferramentas computacionais como as de Processamento de Alto Desempenho. Nesta seção, são analisados os resultados da vetorização via "caminho" de fatoração do algoritmo convencional de Pontos Interiores.

Os algoritmos utilizados foram implementados em Fortran 77 e testados no computador CRAY Y-MP2E/232 do CESUP, da Universidade Federal de Rio Grande do Sul. As mesmas tolerâncias citadas na seção anterior foram utilizadas para a obtenção dos

resultados mostrados a seguir. O desvio na magnitude de tensão permitido foi de 10%. Os resultados numéricos foram obtidos com o sistema teste IEEE 118 barras e com três redes (de 176, 352 e 750 barras) equivalentes do sistema Sul-Sudeste Brasileiro.

O procedimento de vetorização automática utilizado para fins de comparação, consiste no algoritmo tradicional compilado com vetorização automática e otimização escalar. Este procedimento evidencia que os ganhos obtidos com a vetorização proposta são comparados com a “melhor” vetorização automática do computador.

6.6.2. Resultados da Vetorização por Iteração.

A Tabela 4 apresenta os resultados obtidos pelos algoritmos com vetorização automática e “dedicada” através do “caminho” de fatoração. A Figura 15 apresenta os ganhos por iteração e número total de iterações para a convergência.

Tabela 4: Algoritmo Convencional - Vetorização

Sistema	Número de Iterações	Tempo até a Convergência (seg)			Tempo por Iteração (seg)		
		Esc.	Vet.	Speed-up	Esc.	Vet.	Speed-up
118 barras	14	9.28	2.93	3.2	0.56	9.1×10^{-2}	6.2
176 barras	10	4.68	1.53	3.1	0.32	5.9×10^{-2}	5.5
352 barras	14	12.3	3.76	3.3	0.74	1.2×10^{-1}	6.2
750 barras	26	216.1	47.3	4.6	7.87	9.6×10^{-1}	8.2

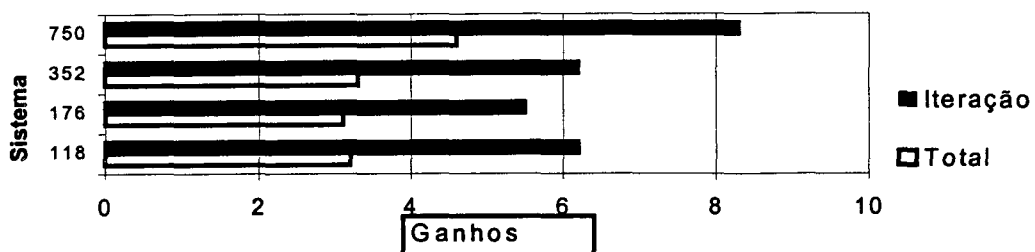


Figura 15: Algoritmo Convencional - Vetorização - Ganhos por Iteração e no Total da Convergência.

Conforme observado na Tabela 4 e na Figura 15, os ganhos com o algoritmo vetorial na convergência variam entre 3 e 4.6, com tempos de cálculo para a implementação vetorial inferiores a 1 minuto. Os ganhos obtidos em uma iteração típica são substancialmente maiores, podendo superar 8 vezes. Esta diferença se origina na primeira iteração, pois nesta são realizadas tarefas que não se repetem nas iterações subseqüentes. Com relação a este aspecto cabe destacar: inicializações, construção da estrutura esparsa e fatoração simbólica. Estas tarefas são eminentemente escalares ou com vetorização automática, reduzindo assim o ganho total do algoritmo.

Numa iteração típica (após a primeira), numerosas operações foram modificadas a fim de permitir a vetorização. Algumas das operações foram estudadas previamente, tais como a vetorização da multiplicação de matrizes esparsas na formação da matriz Hessiana e solução das equações auxiliares.

Os maiores ganhos da implementação vetorial resultam das rotinas de fatoração e substituição direta-inversa do sistema linear, pelo uso do conceito de "caminho de fatoração". Na Tabela 5, apresenta-se o tempo computacional utilizado pelas rotinas e os respectivos "speed-ups", mostrando-se em forma gráfica os ganhos na Figura 16.

Tabela 5: Algoritmo Convencional - Tempos na Fatoração e Substituições com Vetorização Automática e Dedicada.

Sistema	Substituição Direta-Inversa			Fatoração do Sistema Linear		
	Esc.	Vet.	Speed-up	Esc.	Vet.	Speed-up
118 barras	1.3×10^{-2}	3.3×10^{-3}	4.1	0.49	5.6×10^{-2}	9.0
176 barras	1.1×10^{-2}	2.9×10^{-3}	4.0	0.24	2.9×10^{-2}	8.3
352 barras	2.2×10^{-2}	5.2×10^{-3}	4.3	0.46	5.3×10^{-2}	8.7
750 barras	9.5×10^{-2}	2.0×10^{-2}	4.8	6.52	0.67	9.8

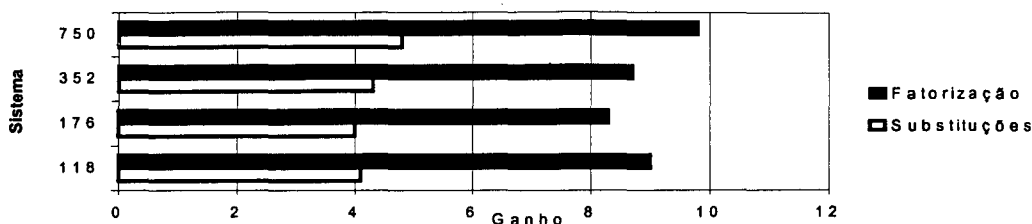


Figura 16: Algoritmo Convencional - Vetorização. Ganhos nas rotinas de Fatoração e Substituição Direta-Inversa.

Conforme se observa na Tabela 5 e na Figura 16, os ganhos relativos às substituições estão na faixa de 4.0 a 4.8. Os tempos computacionais requeridos para a resolução de uma substituição direta-inversa são uma ordem inferior aos necessários para a fatoração. Os ganhos obtidos com a vetorização da rotina de fatoração são superiores, alcançando valores na faixa entre 8.3 e 9.8, devido ao maior número de operações envolvidas.

Os ganhos obtidos são bastante razoáveis para um problema complexo como aquele considerado neste trabalho. A vetorização via "caminho" de fatoração mostrou-se adequada para sua utilização no FPO não-linear, quando utilizada a metodologia de ordenação de Mínimo Grau que considera dinamicamente a condição não-nula do pivô. Nos artigos CASTRONUOVO *et al.* (1998a, 1998b e 1999) são mostrados alguns resultados da vetorização do algoritmo de Pontos Interiores, versão convencional. Em CASTRONUOVO *et al.* (1998b) é apresentada a formulação do método via barreira logarítmica, analisando-se o algoritmo de ordenação modificado e as características de esparsidade da Tabela 2 em sistemas teste e reais de pequeno porte.

No artigo CASTRONUOVO *et al.* (1998a) são consideradas as seguintes modificações: formulação do método de Pontos Interiores convencional através da perturbação das condições de complementaridade e aplicação da vetorização a sistemas reais de medio porte. Neste trabalho o algoritmo de Pontos Interiores Primal-Dual, versão PNL, é apresentado segundo a formulação de (EL-BAKRY *et al.*, 1996; VILLELA, 1997), baseada na perturbação das condições de otimalidade de KKT para o problema original. Esta formulação não requer da adição de funções adicionais para descrever o método, como acontece com a barreira logarítmica utilizada em CASTRONUOVO *et al.* (1998b). A metodologia resultante possui maior clareza conceitual e permite a interpretação dos algoritmos apresentados no Capítulo 4, propostos para estudo neste trabalho. Os resultados obtidos com sistemas de maior porte validam a metodologia empregada, permitindo diminuições apreciáveis nos tempos de cálculo. A redução dos tempos

computacionais favorecem a aplicação da metodologia de vetorização proposta em tempo real.

Em CASTRONUOVO *et al.* (1999) é realizado um aprofundamento na análise da vetorização do algoritmo de Pontos Interiores convencional, acrescentando resultados das principais rotinas na solução do sistema linear.

Os testes com o sistema equivalente sul-sudeste brasileiro de 750 barras apresentam interessantes ganhos em termos de tempo computacional com a vetorização do algoritmo. A melhora nos resultados, em comparação com os testes realizados em sistemas de menor porte, pode ser atribuída ao maior número de operações em ponto flutuante realizadas para a sua resolução, e à maior dimensão do sistema linear a ser resolvido em cada iteração do método (de ordem 3000). Para este sistema realístico, o "speed-up" total até a convergência é de 4.6 e o ganho por iteração é de 8.2.

6.7. Algoritmo de Direções Sucessivas.

O algoritmo de Direções Sucessivas propõe alternar os passos de Centralização e Afim-Escala, dependendo da distância entre a trajetória de busca e o Caminho Central. Se o ponto atual da trajetória de busca se encontra próximo ao Caminho Central dentro de uma determinada tolerância, um passo na direção Afim-Escala é realizado. Caso contrário, um passo de centralização é adotado.

Na Tabela 6 é mostrado o número de iterações de Centralização, Afim-Escala e Total para a convergência do algoritmo de Pontos Interiores de Direções Sucessivas para os sistemas mencionados anteriormente, além daquele correspondente a versão convencional. A distância máxima permitida corresponde ao valor da distância média primal-dual na iteração ($\varepsilon = dmpd$).

Tabela 6: Direções Sucessivas - Número de Iterações.

Sistema	Convencional	Direções Sucessivas		
	Número de Iterações	Número de Iterações		
		A-E	Centr.	Total
IEEE 30 barras	10	6	8	14
IEEE 57 barras	11	8	12	20
IEEE 118 barras	14	8	11	19
SSB 176 barras	10	6	11	17
SSB 352 barras	14	6	13	19
SSB 750 barras	25	19	18	37

Na Tabela 6, observa-se que o algoritmo de Direções Sucessivas requer um número de iterações superior ao da versão convencional para a convergência. A tolerância máxima de afastamento entre a trajetória de busca e o Caminho Central adotada neste exemplo é reduzida, considerando que o valor das distâncias primais-duais decrescem quase uma ordem por iteração. Conforme anteriormente explicado, tolerâncias maiores podem reduzir o número de iterações do algoritmo de Direções Sucessivas, aproximando a sua performance à da versão convencional, e com o custo de não realizar uma adequada centralização da trajetória. Na prática, uma solução de compromisso deve ser adotada entre a centralização da trajetória e o aumento no número de iterações. O algoritmo convencional inclui uma percentagem de centralização em cada iteração (neste caso, 10%), o que resulta num número de iterações superior ao número de iterações Afim-Escala do algoritmo de Direções Sucessivas, e menor ao número de iterações total desta última versão.

Apesar de requerer um maior número de iterações para a convergência, o algoritmo de Direções Sucessivas é interessante quando observado sob o aspecto dos índices de comportamento das direções no processo iterativo dos métodos de Pontos Interiores.

Na Figura 17 é mostrada a variação do parâmetro de combinação das direções σ nas iterações no sistema SSB de 352 barras.

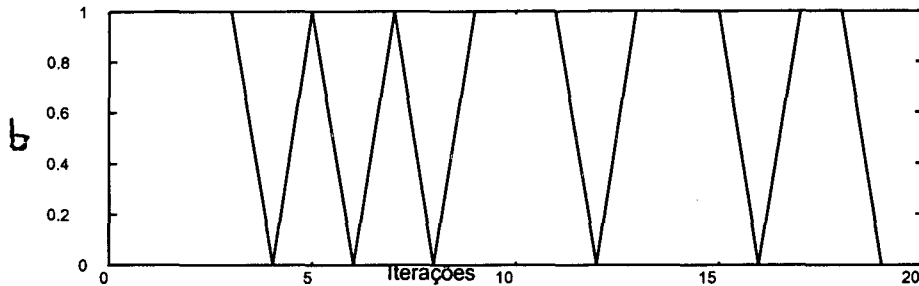


Figura 17: Direções Sucessivas - Variação do σ . SSB 352

Observa-se na Figura 17, que nas primeiras iterações é utilizada a direção de Centralização, a fim de aproximar o caminho de busca ao Caminho Central. Posteriormente, é requerida uma alternância não-sucessiva entre as direções de Centralização e Afim-Escala, devido à diminuição no valor da distância média primal-dual ($dmpd$) e ao afastamento da trajetória de busca do Caminho Central. Na Figura 18 são mostradas as distâncias entre estas trajetórias, para o mesmo sistema, em múltiplos da $dmpd$.

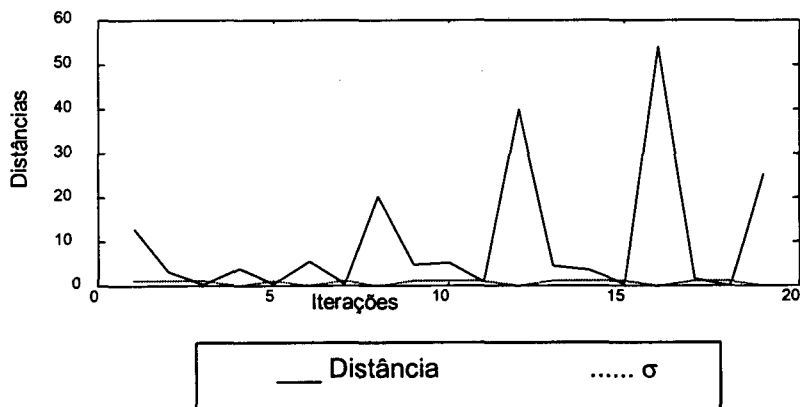


Figura 18: Direções Sucessivas - Variação da Distância ao Caminho Central. SSB 352.

Na Figura 18, representa-se a distância entre a direção de busca e o Caminho Central e o valor de σ no percurso do processo iterativo. Nas primeiras iterações, o algoritmo

procura diminuir a distância ao Caminho Central através de passos de centralização. Na terceira iteração, a tolerância é atingida, realizando-se um passo Afim-Escala. A utilização de um passo Afim-Escala ($\sigma = 0$) numa iteração, aproxima o ponto corrente à solução, apesar de afastar a trajetória de busca do Caminho Central. Por conseguinte, após um passo Afim-Escala são em geral requeridos passos de Centralização. Em algumas circunstâncias, um passo de Centralização não é suficiente para atingir a tolerância adotada (como nas iterações 9, 10 e 11), requerendo-se uma sucessão destas aproximações ao Caminho Central. Entretanto, na Figura 18 observa-se a eficiência dos passos de Centralização na redução dos valores da distância entre as trajetórias de busca e central.

Na Figura 19 é mostrada a variação da distância média primal-dual ($dmpd$) até a convergência. Apresenta-se também nesta mesma figura, a distância entre a direção de busca e o Caminho Central, em valores absolutos, obtidos pela multiplicação dos valores da Figura 18 pelo valor da $dmpd$ na iteração.

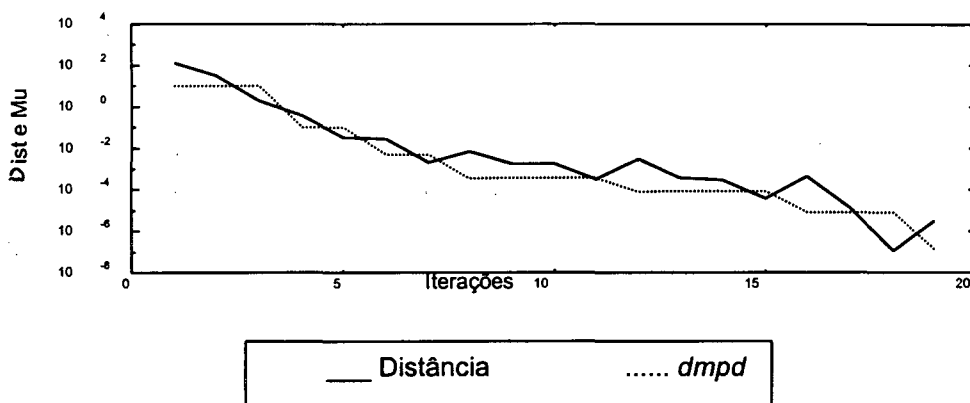


Figura 19: Direções Sucessivas - Variação da $dmpd$. SSB 352.

Neste algoritmo, como nos de Direções Sucessivas Fortemente Centralizado e de Pontos Interiores Centralizado, deve-se utilizar como índice de desempenho a distância média primal-dual ($dmpd$) ao invés de μ . A utilização nos passos de Centralização de $\sigma = 0$ redundava em $\mu = 0$, com problemas na visualização do desempenho do método. Conforme comentado no Capítulo 4, as direções de Centralização não produzem diminuições apreciáveis no valor da $dmpd$. Isto explica as variações não contínuas no decréscimo da $dmpd$ mostradas na Figura 19. De fato, em algumas iterações o valor das distâncias médias pode crescer levemente, como acontece entre as iterações 12 à 15,

devido à imposição de que todas as distâncias primais-duais nas restrições de desigualdade sejam iguais a um valor médio.

A Figura 19 mostra a capacidade do algoritmo de Direções Sucessivas de aproximar a trajetória de busca do Caminho Central. As distâncias entre estas direções vão decrescendo com a diminuição no valor da $dmpd$, produzindo um entrelaçamento entre as curvas. Quando a curva da distância (linha contínua na Figura 19) é inferior ao valor da $dmpd$ (linha pontilhada na mesma figura) um passo na direção Afim-Escala pode ser utilizado. Em caso contrário, a direção de Centralização é utilizada na iteração.

Na Figura 20 é mostrada a variação do máximo valor do desbalanço no gradiente do Lagrangeano durante as iterações.

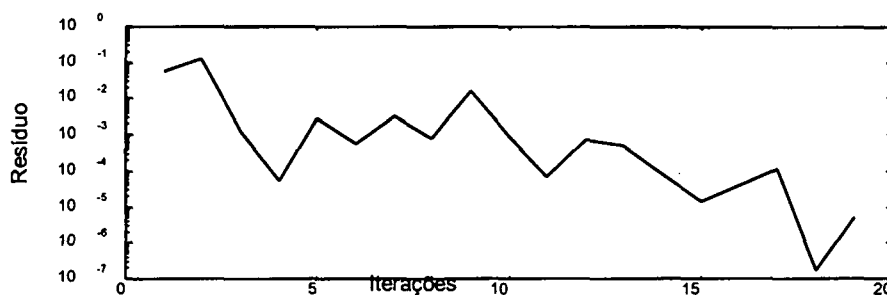


Figura 20: Direções Sucessivas - Variação no Máximo Resíduo.

Como se observa na Figura 20, o máximo resíduo no gradiente do Lagrangeano tem variações bruscas no seu valor, devido às alternâncias entre as direções de Centralização e Afim-Escala. A mudança nos valores de σ e as correspondentes mudanças bruscas na direção de busca provocam as alterações mostradas na Figura 20.

6.8. Algoritmo de Direções Sucessivas Fortemente Centralizado.

O Algoritmo de Direções Sucessivas Fortemente Centralizado é similar à versão implementada na seção anterior. Entretanto, quando é requerida a centralização por encontrar-se a direção de busca afastada da Trajetória Central uma distância δ maior que a tolerância ξ , realiza-se uma série de passos de centralização até atingir uma tolerância menor $\delta < \zeta < \xi$.

O número de iterações de Centralização, Afim-Escala e Total para o algoritmo de Pontos Interiores de Direções Sucessivas Fortemente Centralizado, e para a versão

convencional, com distâncias máximas permitidas de afastamento do caminho central ($\xi = 1 * dmpd$, $\zeta = 0.1 * \xi$), são ilustrados na Tabela 7.

Tabela 7: Direções Sucessivas Fortemente Centralizado – Número de Iterações.

Sistema	Convencional	Direções Sucessivas Fortemente Centralizado		
	Número de Iterações	Número de Iterações		
		A-E	Centr.	Total
IEEE 30 barras	10	6	8	14
IEEE 57 barras	11	8	14	22
IEEE 118 barras	14	8	11	19
SSB 176 barras	10	6	11	17
SSB 352 barras	14	6	14	20
SSB 750 barras	25	19	19	38

Os valores mostrados na Tabela 7 são similares aos encontrados para o algoritmo de Direções Sucessivas. Apenas em alguns sistemas são requeridas iterações adicionais de Centralização. Este algoritmo demonstra a eficiência dos passos de centralização num problema das características do OPF. Poucos passos na direção de Centralização são geralmente suficientes para aproximar a trajetória de busca à Trajetória Central numa dada tolerância. Problemas com maior não-linearidade (VILLELA, 1997), podem requerer iterações adicionais na direção de Centralização.

6.9. Algoritmo de Pontos Interiores Centralizado

O algoritmo de Pontos Interiores Centralizado é similar ao algoritmo de Direções Sucessivas, porém com a direção Afim-Escala substituída pela direção do método de Pontos Interiores convencional. Quando o ponto da iteração atual encontra-se próximo ao

Caminho Central numa dada tolerância, o algoritmo de Pontos Interiores Centralizado realiza uma iteração na direção do algoritmo convencional, a qual inclui uma percentagem de centralização (neste caso, 10%). Se o ponto atual encontra-se afastado do Caminho Central a uma distância superior à tolerância, uma iteração na direção de Centralização é realizada.

Na Tabela 8 é mostrado o número de iterações de centralização, para o método de Pontos Interiores convencional e para o algoritmo de Pontos Interiores Centralizado, com uma distância máxima de afastamento permitida do Caminho Central de $(1 \cdot dmpd)$ e $(\sigma = 0.1)$.

Tabela 8: Pontos Interiores Centralizado - Número de Iterações.

Sistema	Convencional	Pontos Interiores Centralizado		
	Número de Iterações	Número de Iterações		
		PI-Conv.	Centr.	Total
IEEE 30 barras	10	8	6	14
IEEE 57 barras	11	8	11	19
IEEE 118 barras	14	8	9	17
SSB 176 barras	10	6	10	16
SSB 352 barras	14	8	11	19
SSB 750 barras	25	18	16	34

Observa-se na Tabela 8, que o número de iterações total requerido para a convergência do algoritmo de Pontos Interiores Centralizado é inferior, na maioria dos casos, ao requerido pelo algoritmo de Direções Sucessivas (Tabela 6), porém superior ao correspondente ao método convencional. A condição de percorrer uma trajetória na vizinhança do Caminho Central atribui requisitos adicionais ao método de Pontos Interiores, que redundam num maior número de iterações.

O algoritmo de Pontos Interiores Centralizado requer um menor número de iterações que o método de Direções Sucessivas, ao incluir uma percentagem de centralização em todas as iterações. Esta redução é observada principalmente no número de iterações de Centralização.

Na Figura 21 é mostrada a variação no valor de σ , para o sistema SSB 352.

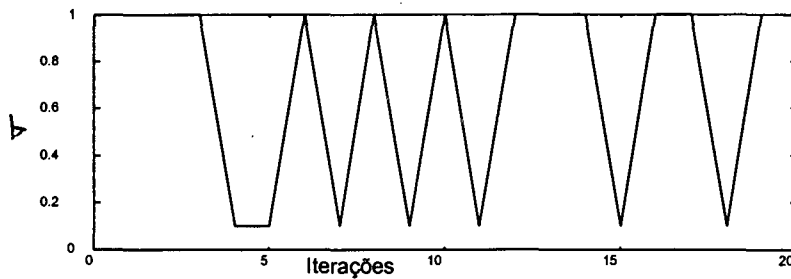


Figura 21: Pontos Interiores Centralizado – Variação do σ .

A Figura 21 mostra a característica do algoritmo de Pontos Interiores Centralizado de alternância não simultânea entre duas percentagens de Centralização, 10 e 100%. Todas as iterações desta versão possuem uma componente de Centralização, diferentemente do algoritmo de Direções Sucessivas. Entretanto, esta componente não é fixa, como acontece na versão convencional do método de Pontos Interiores. Assim, o algoritmo de Pontos Interiores Centralizado combina as características destas versões, fornecendo uma maior flexibilidade ao permitir a variação da percentagem de centralização nas iterações com baixa centralização e a distância de referência entre as direções de busca e o Caminho Central.

A característica de centralização da trajetória do método de Pontos Interiores Centralizado é observada na distância entre a sua trajetória e o Caminho Central, como mostra a Figura 22 para o sistema SSB 352.

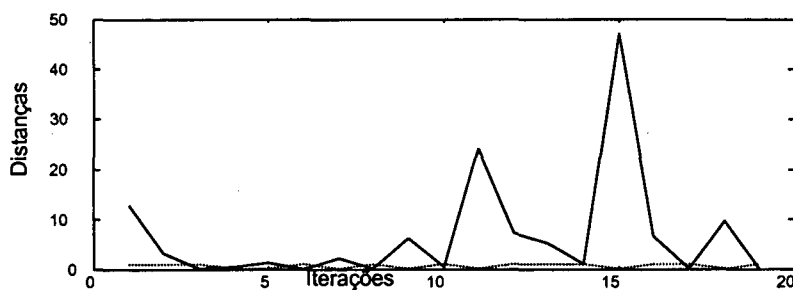


Figura 22: Pontos Interiores Centralizado – Distâncias ao Caminho Central. SSB 352.

Na Figura 22, as distâncias da trajetória ao Caminho Central são medidas em múltiplos de $dmpd$. Observa-se na figura que as características de centralização da trajetória de busca do algoritmo de Pontos Interiores Centralizado são similares àquelas da versão de Direções Sucessivas. Entretanto, um menor número de passos de centralização são requeridos, dado que parte da centralização é realizada nos passos da versão convencional do método.

Uma melhor visualização da variação das distâncias ao Caminho Central no processo iterativo é apresentada na Figura 23. Nesta figura, representa-se os valores de $dmpd$ e as distâncias em valores absolutos nas iterações.

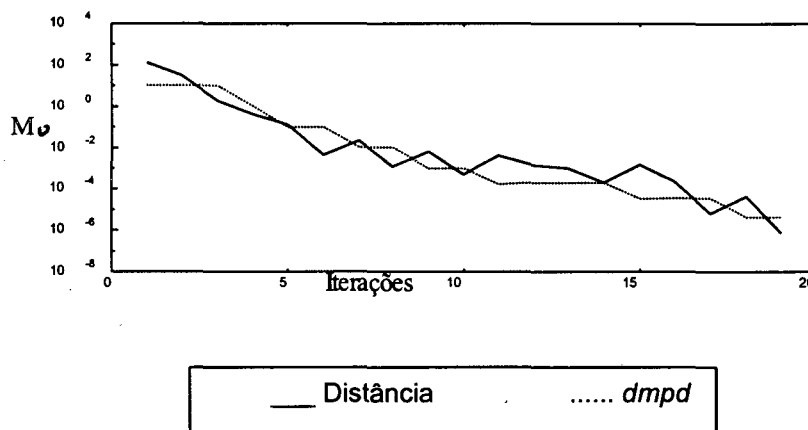


Figura 23: Pontos Interiores Centralizado – Distâncias ao Caminho Central em Valores Absolutos. SSB 352.

A Figura 23 mostra que as distâncias entre a trajetória de busca e o Caminho central decrescem quase continuamente no percurso do processo iterativo, seguindo a diminuição dos valores de $dmpd$. No caso presente, a tolerância permitida entre estas distâncias é o valor de $dmpd$ na iteração. Na Figura 23, observa-se que a curva que representa a distância diminui com os valores de $dmpd$. Da mesma forma que anteriormente, quando é respeitada a tolerância (a linha contínua é inferior à linha pontilhada que representa a $dmpd$) um passo na direção da versão convencional (com $\sigma = 0.1$) é utilizado. Em outro caso, a direção de centralização é adotada.

Na Figura 24 é mostrada a variação do máximo desbalanço para este algoritmo e a sua comparação com a curva para este mesmo índice de desempenho no algoritmo de Direções Sucessivas.

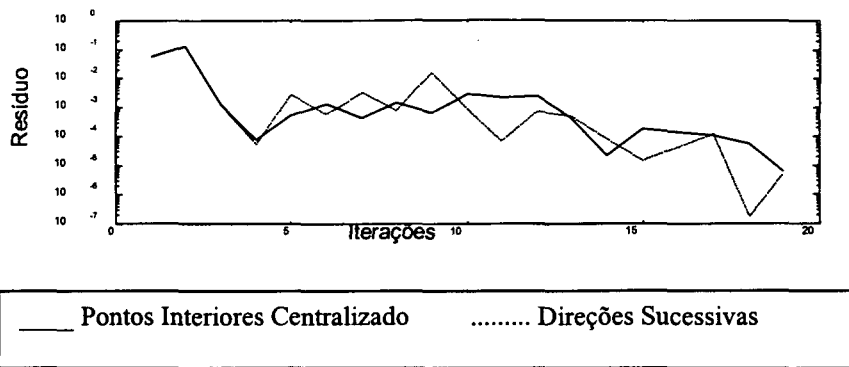


Figura 24: Pontos Interiores Centralizado – Máximo Resíduo.

Na Figura 24 é mostrada a comparação entre o máximo desbalanço no gradiente do Lagrangeano nas versões de Direções Sucessivas e Pontos Interiores Centralizado. A figura mostra a propriedade de “alisamento” na variação dos desbalanço fornecida pela utilização de passos de Pontos Interiores convencional. A utilização de um componente de centralização em todas as iterações permite diminuir a variação brusca nos mistmach apresentada pelo método de Direções Sucessivas.

6.10. Otimização na Direção de Busca.

Esta metodologia consiste em realizar uma pesquisa linear na direção de busca, a fim de se obter um fator de correção que resulte no mínimo valor da distância média primal-dual na iteração corrente. O fator de correção na direção de busca (θ) é obtido quando a distância primal-dual, expressa em função do parâmetro θ , pertence a uma superfície convexa. O valor numérico deste fator deve satisfazer a condição $0 \leq \theta \leq 1$, a fim de manter o sentido da direção de busca do método e a não-negatividade de algumas variáveis.

Na Tabela 9, é mostrado o valor de θ em todas as iterações, para o algoritmo de Pontos Interiores convencional.

Tabela 9: Otimização na Direção de Busca - Valor do θ nas iterações. Algoritmo Convencional.

Iter.	IEEE 30	IEEE 57	IEEE 118	SSB 176	SSB 352	SSB 750
1	1.0	1.0	1.0	1.0	1.0	1.0
2	67	1.0	1.0	1.0	1.0	1.0
3	1000	81	1.0	11	1.0	1.0
4	96	23	190	69	360	310
5	27	20	54	18	999	280
6	29	16	28	7.6	500	390
7	28	68	22	27	120	1.0
8	15	67	33	1.0	8.4	440
9	54	50	31	1.0	4.8	370
10	32	83	13	1.0	4.5	28
11		97	20		25	5.4
12			16		170	7.8
13			37		68	130

14			160		87	8.4
15						8.5
16						1.0
17						7.3
18						10
19						15
20						170
21						52
22						23
23						97
24						1.0
25						1.0

Na Tabela 9, os valores unitários indicam a concavidade da função otimizada na busca linear, o que impede a sua minimização. Para todos os sistemas testados, os valores do fator de correção na otimização na direção de busca encontram-se fora do intervalo $0 \leq \theta \leq 1$. A estreita faixa definida pelos limites de algumas variáveis do FPO (ex.: módulo de tensão nas barras) restringe a variação permitida na direção de busca. Assim, o valor ótimo para a diminuição da distância primal-dual encontra-se, para os sistemas testados, sempre exterior aos limites permitidos para a variação. Este procedimento, que pode melhorar a performance de algoritmos de otimização irrestrita, não apresenta resultados satisfatórios nos testes realizados para a resolução do FPO via Método de Pontos Interiores, tanto para a versão convencional como nas outras versões apresentadas neste trabalho.

6.11. Algoritmo Preditor-Corretor de Mehrotra.

Cada iteração do algoritmo Preditor-Corretor Primal-Dual de Pontos Interiores para PNL realiza duas fases:

Etapa de predição: determinação das distâncias primais-duais na direção Afim-Escala ($\sigma = 0$). A etapa preditora corresponde à solução de um problema não-perturbado ($\mu = 0$).

Etapa de correção: nesta etapa, é resolvida uma substituição direta-inversa adicional e novas equações auxiliares, considerando:

- Expansão das equações de complementaridade até o termo de segunda ordem da série de Taylor;
- Determinação dinâmica do parâmetro de combinação das direções em cada iteração.

Na etapa de correção é obtido o novo ponto na direção de busca.

A Tabela 10 mostra o número de iterações para a convergência do algoritmos Preditor-Corretor e convencional. Os resultados para os sistemas de maior porte são apresentados em CASTRONUOVO *et al.* (2000a e 2000c).

Tabela 10: Preditor-Corretor - Número de Iterações.

Sistema	Convencional	Preditor-Corretor
IEEE 30 barras	10	7
IEEE 57 barras	11	8
IEEE 118 barras	14	10
SSB 176 barras	10	8
SSB 352 barras	14	11
SSB 750 barras	25	18
SSB 1704 barras	21	15

Da Tabela 10, pode ser observada a eficiência do algoritmo Preditor-Corretor na diminuição no número de iterações para a convergência. Na Figura 25, são mostradas as percentagens de redução no número de iterações para este método em relação ao algoritmo convencional.

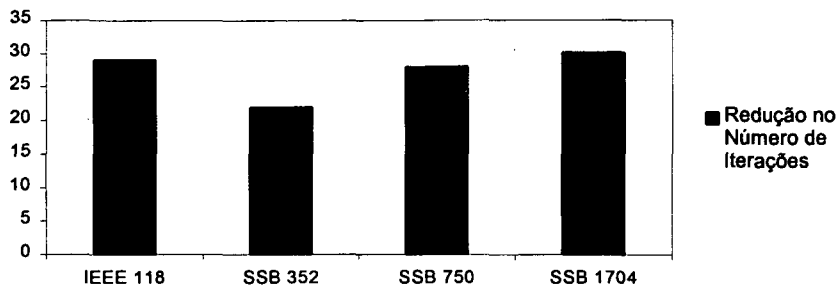


Figura 25: Preditor-Corretor - Percentagens de Redução no Número de iterações. SSB 352.

O número de iterações para o algoritmo Preditor-Corretor é entre 25% e 30% inferior ao da versão convencional do método de Pontos Interiores (Figura 25). Esta redução é devida à melhor representação das equações de complementaridade e à determinação dinâmica do parâmetro de combinação das direções σ .

A Figura 26 representa a variação de σ nas iterações, para o sistema SSB 352.

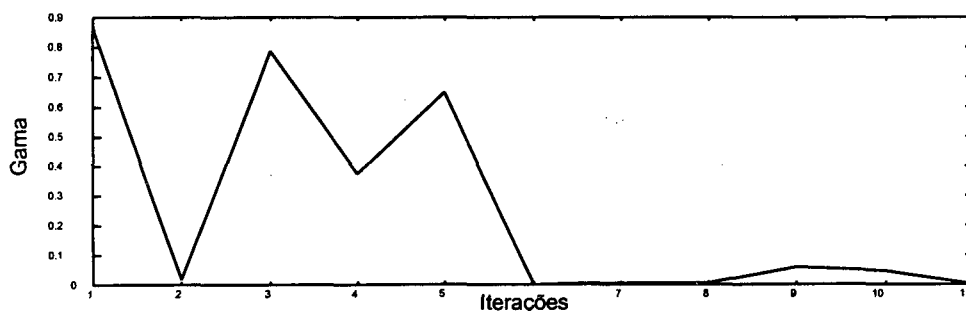


Figura 26: Preditor-Corretor - Variação do σ . SSB 352.

Nos métodos analisados nas seções anteriores deste capítulo, o parâmetro de combinação das direções σ podia assumir um valor constante (método de Pontos Interiores convencional) ou variar entre dois valores pré-fixados (algoritmos de Direções Sucessivas, Direções Sucessivas Fortemente Centralizada e Pontos Interiores Centralizado). No método Preditor-Corretor, o valor de σ pode assumir qualquer valor no

intervalo $(0, 1)$, permitindo ao algoritmo uma maior flexibilidade. Observa-se na Figura 26, que o parâmetro σ varia numa forma não constante no percurso do processo iterativo. No início, uma variação maior nos valores de σ é observada, com altos índices de centralização. Próximo à convergência, o algoritmo utiliza valores baixos de σ , priorizando a convergência do processo através da direção Afim-Escala.

Na Figura 27 é mostrada a variação de μ através do caminho até a convergência.

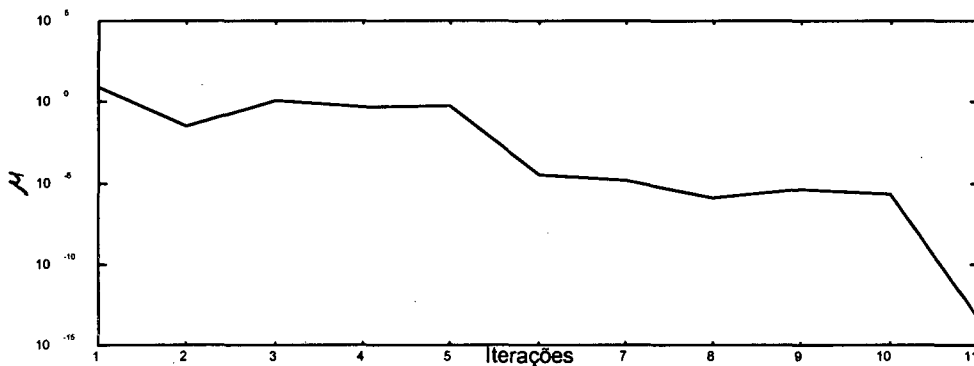


Figura 27: Preditor-Corretor - Variação de μ .SSB 352

Como observado anteriormente com o algoritmo de Direções Sucessivas, o incremento no valor de σ (como acontece na iteração 3) pode redundar num leve aumento no valor da perturbação μ (Figura 27). Deve-se observar entretanto, que a perturbação é o produto das distâncias primais-duais na iteração pelo valor de σ .

As distâncias da trajetória de busca ao Caminho Central são mostradas na Figura 28, para este mesmo sistema.

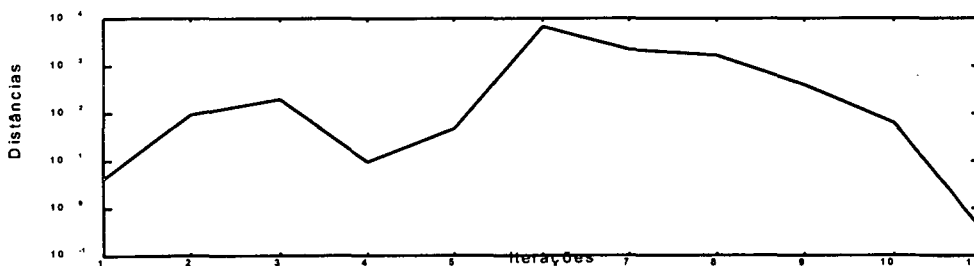


Figura 28: Preditor-Corretor - Distâncias da Trajetória de Busca ao Caminho Central. SSB 352.

Conforme mencionado anteriormente, o algoritmo Preditor-Corretor estima uma trajetória ao ponto candidato a ótimo, visando diminuir o número de iterações até a convergência, através de um esquema de predição e correção da trajetória. Entretanto, a distância da trajetória ao caminho central não é considerada. Por conseguinte, na Figura 28 observa-se que as distâncias entre a trajetória de busca e o Caminho Central variam de iteração a iteração, atingindo valores muito elevados.

Em resumo, o algoritmo Preditor-Corretor procura fornecer uma melhor estimativa para a combinação das direções de centralização e afim-escala. Para isto, é requerido o conhecimento do valor das distâncias primais-duais via direção afim-escala, o que demanda duas substituições direta-inversa do sistema linear em cada iteração. Este esforço computacional adicional torna atrativa a utilização de ferramentas de Alto Desempenho na sua resolução.

6.12. Vetorização do Algoritmo Preditor-Corretor.

Nesta seção é analisada a vetorização do algoritmo Preditor-Corretor e a sua performance comparada a do algoritmo convencional.

Os resultados mostrados nesta seção foram obtidos num computador Cray T94, do Centro Nacional de Supercomputação da UFRGS. O computador Cray T94 é mais moderno que o Cray Y-MP2E/232, utilizado para a avaliação da vetorização do algoritmo convencional de Pontos Interiores. A mudança de plataforma computacional evidenciou os ganhos na vetorização dedicada. Nesta seção, são analisados os ganhos na nova plataforma para os algoritmos Preditor-Corretor e convencional.

Os resultados da Vetorização Dedicada (VD) através do conceito de NCF são comparados com aqueles obtidos com a Otimização Escalar e Vetorização Automática (VA). A Tabela 11 mostra os tempos de processamento na vetorização da substituição direta-inversa e da fatorização, com e sem o uso do conceito de NCF. A Figura 29 apresenta os ganhos obtidos para os maiores sistemas testados.

Tabela 11: Preditor-Corretor – Vetorização. Tempos na Fatoração e Substituição Direta-Inversa (seg.).

Sistema	Substituição		Fatorização	
	VA	VD	VA	VD
IEEE 118	8.09×10^{-3}	1.29×10^{-3}	.194	1.17×10^{-2}
SSB 352	1.34×10^{-2}	1.91×10^{-3}	.170	1.23×10^{-2}
SSB 750	6.16×10^{-2}	7.52×10^{-3}	2.31	.152
SSB 1704	1.73×10^{-1}	1.89×10^{-2}	9.95	.577

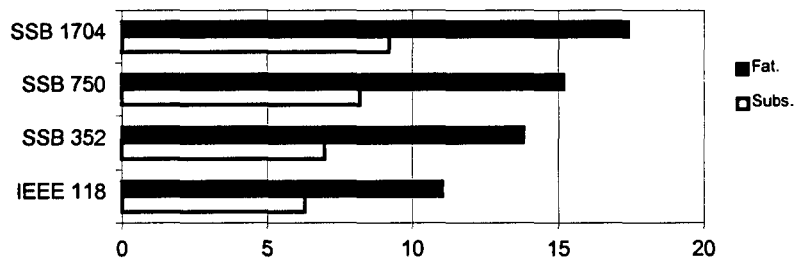


Figura 29: Preditor-Corretor - Vetorização. Ganhos na Fatoração e Substituição Direta-Inversa

Os ganhos na vetorização da fatoração pelo conceito de NCF variam entre 11 e 17, em relação à melhor vetorização automática realizada pelo computador. A diferença nos ganhos na vetorização das rotinas com a mudança da plataforma computacional evidenciam as melhoras inerentes ao equipamento, dado que não foram realizadas modificações no algoritmo. No computador Cray Y-MP2E/232, para os sistemas maiores, são obtidos ganhos na ordem de 9 vezes. O computador Cray T94 eleva estes ganhos em aproximadamente 70% para a mesma rotina. Estes ganhos dizem respeito tanto no algoritmo convencional como no Preditor-Corretor, pois ambos os algoritmos requerem a realização de fatoração.

Na substituição direta-inversa, os ganhos encontram-se na faixa 6.3 e 9.2 para o computador Cray T94. Estes ganhos são também aproximadamente 70% maiores que os obtidos no Cray Y-MP2E/232. Esta melhoria nos ganhos das rotinas de substituição direta-inversa são relativas principalmente às opções do algoritmo de Pontos Interiores, as quais requerem em geral de duas destas operações por iteração.

Como anteriormente, o maior número de operações executadas na fatorização redundam em ganhos superiores àqueles obtidos na substituição direta-inversa.

Os algoritmos convencional e Preditor-Corretor demandam uma e duas substituições direta-inversa, respectivamente, a cada iteração. Esta substituição extra, no algoritmo Preditor-Corretor, resulta em tempos computacionais maiores por iteração. As tabelas seguintes apresentam os tempos computacionais para os algoritmos convencional e Preditor-Corretor, considerando as vetorizações dedicada e automática. A Tabela 12 fornece o número de iterações e os tempos de cálculo até convergência, e a Tabela 13 mostra os resultados por iteração.

Tabela 12: Preditor-Corretor – Vetorização. Tempos até a Convergência (seg.).

Sistema	Convencional			Preditor-Corretor		
	Iter.	VA	VD	Iter.	VA	VD
IEEE 118	14	5.07	1.78	10	4.31	1.52
SSB 352	14	5.43	2.00	11	5.01	1.91
SSB 750	25	83.8	23.3	18	74.3	21.3
SSB 1704	21	328	80.3	15	298	74.2

Tabela 13: Preditor-Corretor – Vetorização. Tempos por Iteração (seg.).

Sistema	Convencional		Preditor-Corretor	
	VA	VD	VA	VD
IEEE 118	.219	3.3×10^{-2}	.245	3.4×10^{-2}
SSB 352	.248	4.5×10^{-2}	.286	4.9×10^{-2}
SSB 750	2.76	2.7×10^{-1}	3.27	2.9×10^{-1}
SSB 1704	10.7	9.5×10^{-1}	12.1	9.9×10^{-1}

A mudança na plataforma computacional reduz, para o algoritmo convencional, quase à metade os tempos computacionais do processo iterativo nos sistemas testados. Como

exemplo, o problema de FPO com vetorização dedicada no sistema SSB de 750 barras é resolvido no computador Cray Y-MP2E/232 em 47.3 segundos, e o computador Cray T94 diminui este valor a 23.3 segundos.

No algoritmo Preditor-Corretor, a vetorização através do conceito de NCF permite obter ganhos por iteração entre 6 e 12 nos sistemas testados. Os ganhos no total do processo iterativo, sempre para o mesmo algoritmo, são menores e variam entre 2.6 e 4.1. Como anteriormente, as diferenças entre os ganhos por iteração e para o processo iterativo surgem no processo de inicialização e na primeira iteração. Estas etapas consistem principalmente de operações com vetorização automática (inicializações) ou essencialmente escalares (ordenação, fatorização simbólica, etc.), que reduzem os ganhos dos algoritmos.

A cada iteração, o algoritmo Preditor-Corretor requer maior quantidade de operações do que a versão convencional. Estas operações adicionais redundam em tempos computacionais mais elevados. Na Figura 30, são comparados os incrementos nos tempos de cálculo por iteração entre os algoritmos para as duas opções de vetorização, em percentagem. Por exemplo, o incremento no tempo por iteração do algoritmo Preditor-Corretor em relação ao algoritmo convencional, usando o conceito de NCF, é calculado através de $Incremento = \frac{(T_{VD}^{P-C} - T_{VD}^C)}{T_{VD}^C} * 100$, onde T_{VD}^C e T_{VD}^{P-C} são os tempos por iteração nos algoritmos convencional e Preditor-Corretor, com Vetorização Dedicada.

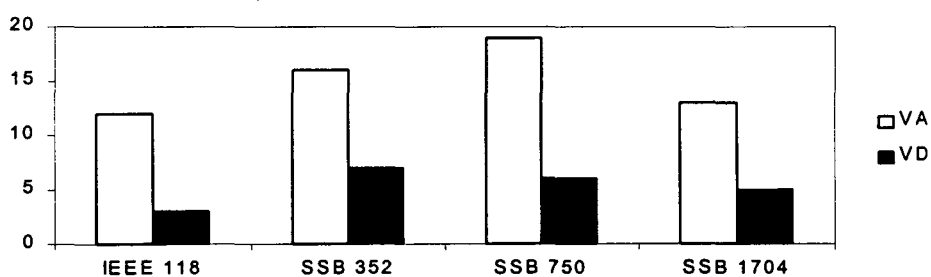


Figura 30: Preditor-Corretor – Vetorização. Incrementos nos Tempos por Iteração (percentagens).

Conforme observado na Figura 30, os tempos por iteração para o algoritmo Preditor-Corretor são entre 12% e 18% superiores aos requeridos pelo algoritmo convencional, se for usada a Vetorização Automática. A utilização do conceito de NCF permite reduzir o tempo demandado pelas operações adicionais a valores inferiores a 7%. Deve-se

ressaltar, que entre as operações adicionais encontram-se a adição de uma substituição direta-inversa e a resolução das equações auxiliares, procedimentos nos quais é utilizada a vetorização via NCF.

Como anteriormente expresso, o algoritmo Preditor-Corretor requer operações adicionais que redundam em tempos computacionais superiores por iteração. Entretanto, ele é eficiente, em geral, na diminuição no número de iterações para a convergência. Um balanço entre estas propriedades é mostrado na Figura 31, em função dos tempos computacionais no processo iterativo.

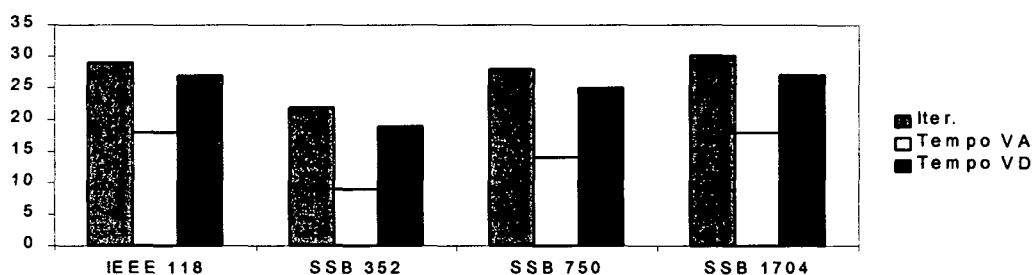


Figura 31: Preditor-Corretor – Vetorização. Redução no Número de iterações e nos Tempos do Processo Iterativo (percentagens).

Na Figura 31, é observada uma redução considerável (aproximadamente 30%) no número de iterações requeridas pelo algoritmo Preditor-Corretor, devidas à melhoria na determinação da trajetória de busca da solução ótima. Porém, a redução no número de iterações não é refletida por completo na diminuição dos tempos computacionais (Tabela 12). Duas são as razões principais para isto: a) a existência de operações não-vetoriais ou com vetorização automática no início do algoritmo; e b) o aumento dos tempos por iteração.

Os efeitos da vetorização podem ser melhor avaliados se forem suprimidos os efeitos das operações iniciais no processo iterativo. O produto (*Número_de_Iterações x Tempo_por_Iteração*) fornece uma medida do esforço computacional requerido no processo iterativo de cada algoritmo. No processo iterativo do algoritmo Preditor-Corretor com Vetorização Automática, as reduções nos tempos computacionais são em torno de 15%, diminuindo à metade as reduções conseguidas no número de iterações. Entretanto, as reduções de tempo computacional obtidas pela aplicação do conceito de NCF são semelhantes (média 4% inferiores) àquelas obtidas no número de iterações (Figura 31). A

vetorização proposta é eficiente no aproveitamento da redução no número de iterações obtidas com o algoritmo Preditor-Corretor, melhorando a eficiência computacional do método.

Nos trabalhos CASTRONUOVO *et al.* (2000a e 2000c) é analisada a vetorização do algoritmo Preditor-Corretor através dos níveis do "caminho" de fatoração. Em CASTRONUOVO *et al.* (2000a) é considerada a vetorização do algoritmo Preditor-Corretor e a sua comparação com o algoritmo convencional. Resultados dos testes nos sistemas IEEE 118, SSB 352, SSB 750 e SSB 1704 são mostrados, destacando-se as propriedades de vetorização do algoritmo. São mostradas nesse trabalho as reduções no tempo computacional fornecidas pela aplicação da vetorização em ambos algoritmos. Em CASTRONUOVO *et al.* (2000c) é realizado um aprofundamento na análise da vetorização do algoritmo Preditor-Corretor, mostrando-se as principais propriedades da aplicação. A redução no número de iterações obtida com o algoritmo é considerada, mostrando-se as diferenças com os tempos computacionais requeridos para a convergência.

Observa-se que o algoritmo Preditor-Corretor mostra-se eficiente na redução no número de iterações requerido para a resolução de um FPO não-linear. A adequada utilização de Processamento de Alto Desempenho permite que esta redução seja expressa nos tempos computacionais até a convergência. Por conseguinte, esta versão do método de Pontos Interiores pode ser competitiva com o algoritmo convencional para aquelas aplicações que requerem de tempos de processamento reduzido.

Entretanto, a não consideração da distância ao Caminho Central no algoritmo Preditor-Corretor pode produzir problemas na convergência de sistemas com variáveis com reduzido intervalo de variação.

6.13. Algoritmo Preditor-Corretor com Múltiplas Correções.

O algoritmo Preditor-Corretor com Múltiplas Correções inicia em cada iteração com uma predição na direção Afim-Escala (sem perturbação). Na etapa de correção é verificada se há necessidade da centralização na iteração, aproximando-se a equação de complementaridade por uma função de segundo grau. Enquanto a versão do método de Pontos Interiores Preditor-Corretor realiza só uma correção por iteração, o algoritmo Preditor-Corretor com Múltiplas Correções requer mais de uma correção na iteração. A realização de correções adicionais depende de dois aspectos: 1) se a correção decresce efetivamente o produto primal-dual, e 2) o número máximo de correções permitidas.

Na Tabela 14 é mostrado o número de iterações para a convergência das versões convencional, Preditor-Corretor e Preditor-Corretor com Múltiplas Correções. Para este último método são mostrados os resultados para um número máximo de 2, 3, 4 e 5 correções permitidas por iteração.

Tabela 14: Preditor-Corretor com Múltiplas Correções - Número de Iterações.

Sistema	Conv	P-C	P-C Múltiplas Correções							
	Iter	Iter	2		3		4		5	
			Iter	Corr.	Iter	Corr.	Iter	Corr.	Iter	Corr.
IEEE 30	10	7	7	14	7	20	6	22	6	27
IEEE 57	11	8	9	16	9	19	9	22	9	24
IEEE 118	14	10	9	17	8	21	8	23	8	30
SSB 176	10	8	7	13	8	17	8	19	8	23
SSB 352	14	11	10	19	9	23	10	25	NC	-
SSB 750	25	18	18	31	53	91	NC	-	NC	-

Na Tabela 14, NC representa a não convergência do algoritmo após 100 iterações. O sistema IEEE 57 barras (Tabela 14) é o único para o qual o número de iterações aumenta ao se permitir mais de uma correção por iteração. Para o sistema SSB 750 o aumento no número de correções não reduz o número de iterações, o que é atribuído às suas características de acentuada não-linearidade. Nos outros sistemas, o aumento no número de correções realizadas por iteração decresce o número de iterações requeridas para a convergência. As reduções no número de iterações variam desde 12.5% até 20% para estes sistemas, se comparadas com o algoritmo Preditor-Corretor com apenas uma correção. Deve-se notar que o aumento no número de correções pode trazer problemas de convergência, como mostrado na Tabela 14.

Observa-se que não há um comportamento bem definido da convergência deste algoritmo com relação ao número de correções adotado. Os menores números de iterações para a convergência foram obtidos, para os diferentes sistemas, com distintos valores do número de correções. Um dos sistemas requer uma correção adicional por iteração, dois outros sistemas requerem um máximo de duas correções, e para um outro sistema o menor número de iterações é alcançado com quatro correções por iteração.

Considere-se o sistema SSB 352 barras, para a análise do algoritmo Preditor-Corretor com Múltiplas Correções. Com apenas uma correção (versão Preditor-Corretor), a convergência é alcançada em 11 iterações, 3 a menos do que com o algoritmo convencional. A realização de uma correção adicional em cada iteração reduz uma unidade o número de iterações para convergência. Para se obter essa redução, em 90% das iterações realiza-se essa correção adicional. Se até 3 correções por iteração são adotadas, o número de iterações é reduzido em outra unidade, fornecendo uma diminuição total no número de iterações para a convergência de 18,2% em relação ao método Preditor-Corretor com uma correção. A realização de quatro correções por iteração piora as características de convergência do método neste sistema, e com cinco correções por iteração o método não converge.

Na Figura 32, é mostrada a variação do parâmetro de combinação das direções no percurso do processo iterativo, para o sistema SSB 352, com um máximo de três correções por iteração. Para fins de comparação, também é apresentada a curva de variação do parâmetro para o método Preditor-Corretor com uma correção.

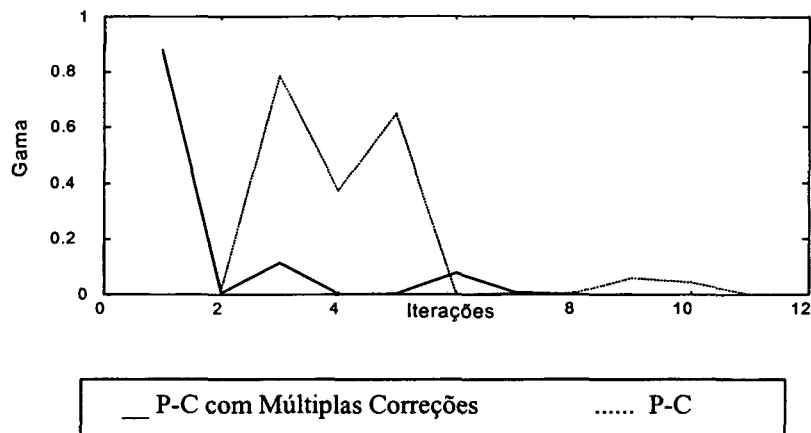


Figura 32: Preditor-Corretor com Múltiplas Correções - Variação do σ .

Como se observa na Figura 32, a realização de mais de uma correção nas iterações resulta na redução no valor de σ . A representação das equações de complementaridade por curvas quadráticas e a resolução mais de uma vez por iteração desta aproximação, permitem utilizar percentagens menores de centralização na trajetória de busca.

Na Figura 33 é representada a distância da trajetória de busca em relação ao Caminho Central nas iterações.

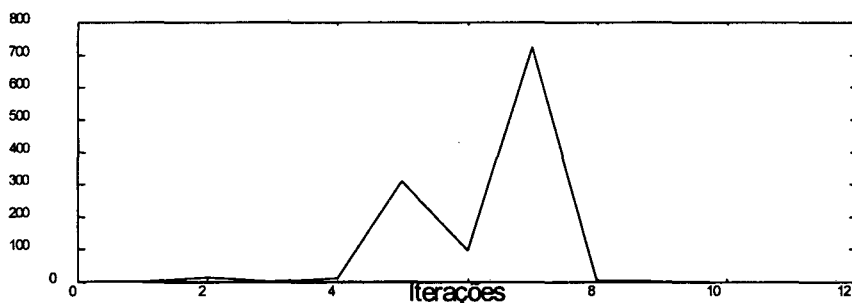


Figura 33: Preditor-Corretor com Múltiplas Correções - Variação da Distância.

É observado na Figura 33, que o método Preditor-Corretor com Múltiplas Correções, da mesma forma que a versão Preditor-Corretor com apenas uma correção, não leva em consideração a proximidade ao Caminho Central. Nas iterações nas quais poucas correções são efetuadas, a aproximação ao Caminho Central é prejudicada, resultando nos valores elevados mostrados na Figura 33.

Os resultados apresentados mostram a eficiência do algoritmo Preditor-Corretor com Múltiplas Correções na redução do número de iterações para a convergência da maioria dos sistemas testados. Entretanto, as operações adicionais requeridas nas correções auxiliares (substituição direta-inversa e equações auxiliares) reduzem a eficácia computacional do método. Observa-se ainda, problemas de convergência nos sistemas maiores com o acréscimo de correções auxiliares.

6.14. Máximo Passo na Trajetória Central

6.14.1. Resultados da Convergência.

O algoritmo de Máximo Passo na Trajetória Central visa percorrer uma trajetória próxima ao Caminho Central no seu caminho até a convergência. Em cada iteração do processo iterativo, são calculadas as direções de Centralização e Afim-Escala. Em função da máxima distância de afastamento permitida ao Caminho Central, é determinado um parâmetro σ que combina as direções encontradas a fim de satisfazer a condição de vizinhança à trajetória central. Assim, uma combinação linear em σ das direções de Centralização e Afim-Escala fornece o próximo ponto na trajetória de busca.

Como acontece no método Preditor-Corretor, o valor de σ em cada iteração é calculado, segundo um critério pré-estabelecido. O método Preditor-Corretor visa diminuir a perturbação μ , iteração a iteração, através de uma trajetória próxima à direção Afim-Escala, diminuindo o número de iterações. Por outro lado, o algoritmo de Máximo Passo na Trajetória Central procura uma trajetória de busca na vizinhança do Caminho Central. As características de convergência e o caminho percorrido são então diferentes para estes algoritmos.

Na Tabela 15 são mostrados os resultados de convergência do algoritmo de Máximo Passo na Trajetória Central, para o problema de FPO formulado anteriormente. A tolerância para o máximo desbalanço de potência, o vetor gradiente e o parâmetro de perturbação (μ) é 10^{-7} . A distância máxima permitida de afastamento da trajetória central é ($3*\mu$). O algoritmo foi implementado em Fortran 90 e os resultados numéricos obtidos num computador Pentium II de 400 Mhz com 128b de memória Ram (CASTRONUOVO *et al.*, 2000b).

Tabela 15: Máximo Passo - Tempos até Convergência (seg.).

Sistema	Convencional		Máximo Passo	
	Iter.	Tempo	Iter.	Tempo
IEEE 30	10	0.21	10	0.27
IEEE 57	12	0.59	12	0.71
IEEE 118	16	2.18	18	2.75
SSB 176	16	1.63	18	2.03
SSB 352	14	3.20	18	4.48
SSB 750	27	12.9	29	15.8

A solução do problema de otimização via algoritmo de Máximo Passo na Trajetória Central requer tempos computacionais superiores aos obtidos com o método convencional (média de 26% superiores). Isto é devido não apenas a um acréscimo no número de iterações para os sistemas maiores, mas também à existência de operações adicionais por iteração. Fundamentalmente, o aumento no tempo de cálculo por iteração provém da substituição direta-inversa adicional requerida pelo procedimento de Máximo Passo na Trajetória Central. Os tempos computacionais requeridos por dita operação podem ser reduzidos mediante a aplicação de Técnicas Computacionais de Alto Desempenho. Não são apresentados no presente trabalho, os resultados da vetorização do método de Máximo Passo na Trajetória Central, por serem similares aos obtidos para o algoritmo Preditor-Corretor.

As figuras seguintes mostram as variações dos principais índices de desempenho no processo iterativo do algoritmo de Máximo Passo na Trajetória Central para o sistema SSB 352 barras. São representados os valores do máximo resíduo (Figura 34), da função objetivo (Figura 35), da perturbação μ (Figura 36) e do parâmetro de combinação das direções σ (Figura 37).

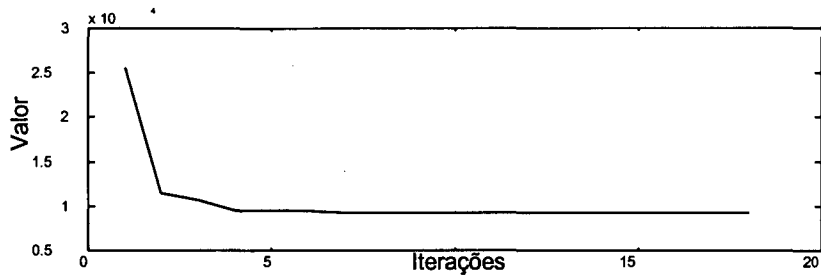


Figura 34: Máximo Passo na Trajetória Central - Valor do Máximo Resíduo.

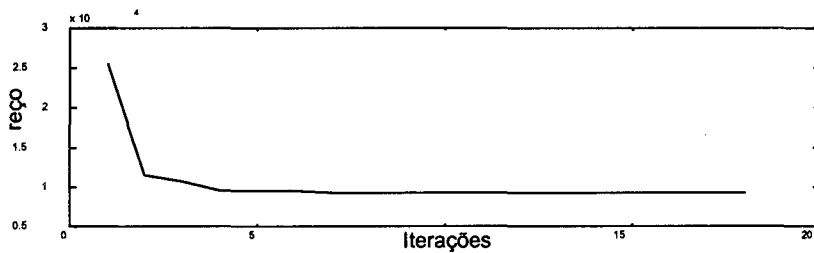


Figura 35: Máximo Passo na Trajetória Central - Valor da Função Objetivo.

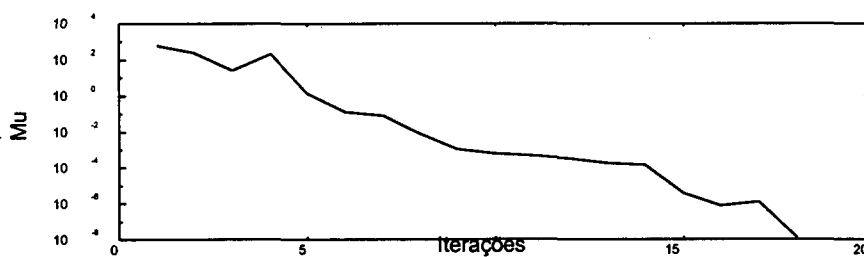


Figura 36: Máximo Passo na Trajetória Central - Valor da Perturbação.

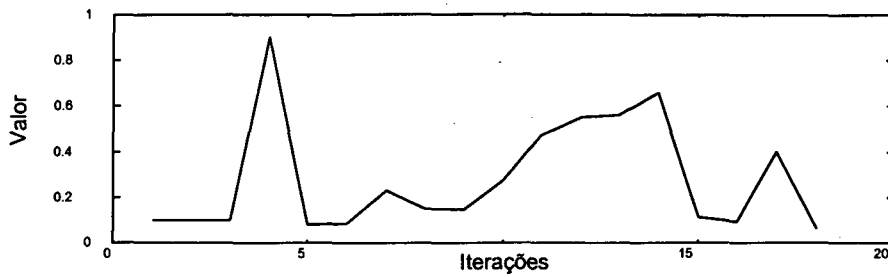


Figura 37: Máximo Passo na Trajetória Central - Valor do Parâmetro de Combinação das Direções.

Nas Figura 34 e 35, são observadas variações monotônicas no máximo resíduo e no valor da função objetivo no percurso do processo iterativo. Entretanto, o valor de μ varia bruscamente em algumas iterações (Figura 36), devido às variações no valor de σ (Figura 37). Como foi dito anteriormente, o algoritmo de Máximo Passo na Trajetória Central determina uma direção de busca próxima ao caminho central, variando o valor de σ a fim de garantir a proximidade esperada ao Caminho Central. O algoritmo possui a capacidade de combinar automaticamente as direções Afim-Escala e de Centralização, calculando um valor de σ adequado a esta distância. Na Figura 37, observa-se que nas primeiras iterações o algoritmo não consegue garantir a vizinhança solicitada, utilizando um valor de σ pré-especificado (neste trabalho, 0.1), satisfazendo nas restantes iterações o valor esperado de proximidade ao caminho central.

A principal característica do algoritmo de Máximo Passo na Trajetória Central é mostrada na Figura 38.

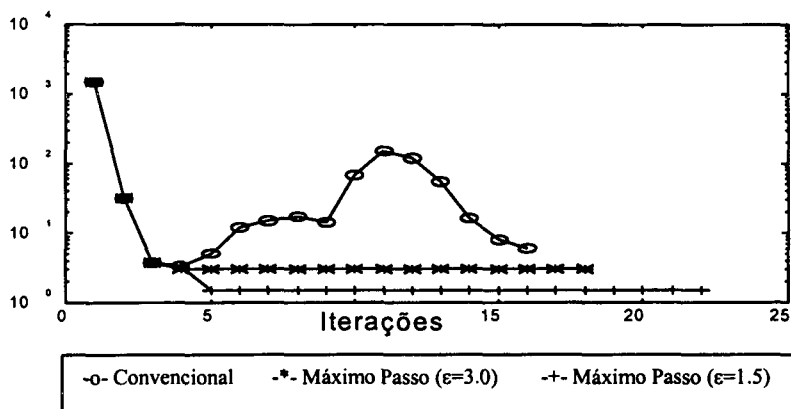


Figura 38: Máximo Passo na Trajetória Central – Distâncias ao Caminho Central.

Na Figura 38, são mostradas as normas quadráticas das distâncias da trajetória percorrida até a convergência pelo método convencional e pelo algoritmo de Máximo Passo na Trajetória Central. O primeiro destes métodos não consegue acompanhar

adequadamente o caminho central, com uma trajetória de busca que em determinados momentos se afasta consideravelmente da trajetória central. Entretanto, o algoritmo de Máximo Passo na Trajetória Central é eficiente em garantir uma distância máxima pré-especificada ao caminho central. Após as primeiras iterações, onde o algoritmo não consegue encontrar um σ adequado e prossegue pela trajetória da versão convencional, uma trajetória tão central quanto for especificada é percorrida. Na Figura 38, é observado o comportamento do algoritmo para dois valores de proximidade, $3.0*\mu$ e $1.5*\mu$. É também possível observar que esta característica de centralização redundando num número de iterações superior, tanto maior quanto menor seja o valor de proximidade especificado.

6.15. Aplicação do Algoritmo de Máximo Passo na Trajetória Central

O algoritmo de Máximo Passo na Trajetória Central percorre uma trajetória de convergência vizinha ao Caminho Central. Por conseguinte, pode resultar conveniente a sua utilização em regiões de convergência demasiadamente restritas, com pequenos intervalos de variação permitida nas variáveis de otimização. Nesta seção, é mostrado um exemplo de aplicação do algoritmo Máximo Passo na Trajetória Central neste tipo de região.

6.15.1. Minimização das Perdas.

Em CASTRONUOVO *et al.* (2000b), é proposta a resolução de um problema de redução de perdas no sistema elétrico, com o objetivo de testar o algoritmo de Máximo Passo na Trajetória Central em problemas com região das soluções viáveis restrita. No caso estudado, a função objetivo é a minimização da geração de potência ativa na barra de referência, com potências ativas pré-especificadas nas outras barras de geração. A região das soluções viáveis é modificada pela redução nos intervalos de variação das potências ativas geradas na vizinhança de valores ótimos obtidos previamente. A solução ótima obtida num FPO de minimização do custo de geração foi adotada como geração pré-especificada. As variações permitidas para a geração de potência ativa são entre ± 0.1 p.u. Na Tabela 16, é mostrado o número de iterações e os tempos computacionais para a obtenção da solução deste problema.

Tabela 16: Máximo Passo – Mínima Perda. Tempos até Convergência (seg.).

Sistema	Convencional		Máximo Passo	
	Iter.	Tempo	Iter.	Tempo
IEEE 30	11	0.22	10	0.22
IEEE 57	11	0.55	9	0.52
IEEE 118	10	1.32	8	1.30
SSB 176	11	1.05	9	0.98
SSB 352	10	2.12	8	1.98

Os resultados da Tabela 16 foram obtidos com os mesmos recursos computacionais que os da Tabela 15. Observa-se que o número de iterações do método de Pontos Interiores convencional (Tabela 16) é geralmente inferior que o requerido no problema de mínimo custo de geração de potência ativa (Tabela 15). Isto é devido principalmente à redução no número das variáveis do problema. No problema de redução das perdas, a potência ativa no barra de referência é a única geração que pode ser efetivamente modificada no problema de otimização.

A modificação efetuada no espaço de soluções enfatiza a eficiência do algoritmo de Máximo Passo na Trajetória Central em percorrer uma trajetória próxima ao caminho central. Da Tabela 16, é observado que a aplicação deste algoritmo resulta em número de iterações e tempos computacionais inferiores aqueles obtidos via método convencional. Na Figura 39, são mostradas as distâncias ao caminho central para ambos algoritmos, no sistema SSB 352.

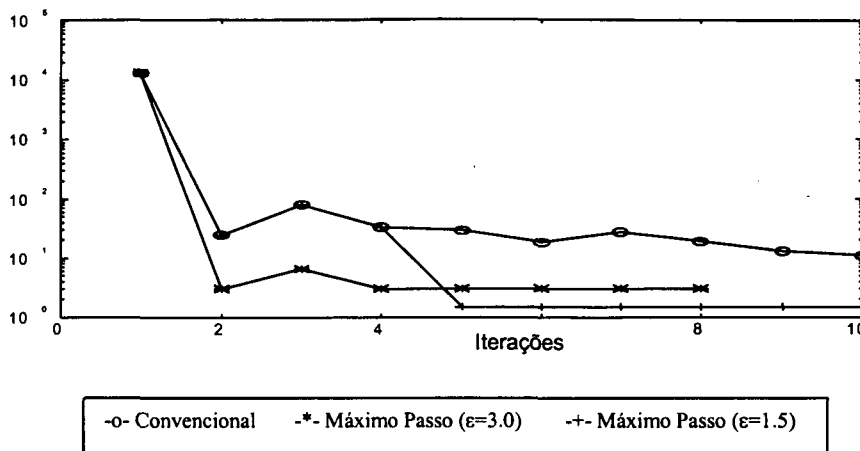


Figura 39: Máximo Passo na Trajetória Central – Mínima Perda. Distâncias ao Caminho Central.

A Figura 39 mostra que o algoritmo de Máximo Passo na Trajetória Central possui a capacidade de percorrer uma trajetória próxima ao Caminho Central, ainda para problemas com regiões de convergência mais restritas. Para uma distância máxima de $1.5*\mu$, o algoritmo requer 5 iterações até encontrar um valor de $\sigma \in (0,1)$ que combine as direções de Centralização e Afim-Escala numa forma adequada. Para uma tolerância maior ($3.0*\mu$), apenas na primeira e na terceira iteração não é encontrado um valor de σ , utilizando-se nestas iterações $\sigma = 0.1$, como no algoritmo convencional.

6.16. Conclusões

Neste capítulo é mostrada a aplicação das versões do método de Pontos Interiores, apresentadas no Capítulo 4, a um problema de FPO. É também considerada a utilização de Programação de Alto Desempenho naqueles algoritmos com maior potencial.

Os resultados apresentados são mostrados numa forma comparativa ao algoritmo convencional, destacando-se as suas particularidades teóricas e de comportamento no percurso do processo iterativo. As características de convergência dos algoritmos mostraram-se fortemente dependentes do critério utilizado para a combinação das direções de Centralização e Afim-Escala, modificando o número de iterações para a convergência e a robustez do algoritmo.

Resultados são mostrados em sistemas teste e realísticos, de médio e grande porte, desde 30 até 1700 barras ilustram a performance destes algoritmos.

Capítulo 7.

Conclusões e Sugestões Para Futuros Trabalhos

7.1. Introdução

O método de Pontos Interiores possui boas características de convergência, combinando um adequado tratamento das restrições de desigualdade à utilização das informações das derivadas de segunda ordem das funções envolvidas no problema de otimização. Sua performance na resolução de problemas de otimização restritos motivaram a pesquisa na área, principalmente na programação linear.

Em cada iteração do algoritmo de Pontos Interiores obtém-se uma direção de busca, a qual é combinação linear de três direções: de Viabilidade das Restrições de Igualdade, de Centralização da Trajetória e Afim-Escala. Em particular, as duas últimas direções são ajustadas através de um parâmetro de combinação. Distintas formas de combinação das direções constituem as variadas versões do método de Pontos Interiores, dependendo a performance de cada versão do critério utilizado para a combinação.

Neste trabalho, algumas destas versões foram analisadas, estudando-se a sua aplicabilidade ao problema de FPO. As conclusões obtidas são apresentadas na seção seguinte.

7.2. Conclusões do Trabalho

Da análise teórica dos assuntos abordados e dos resultados numéricos obtidos da implementação computacional, as seguintes observações podem ser feitas:

- A pesquisa bibliográfica mostra um número apreciável de trabalhos sobre métodos de Trajetória Central, devido às suas boas características de convergência. Esta pesquisa é principalmente orientada a problemas de otimização linear, pela facilidade matemática na demonstração das propriedades de convergência;
- O método de Pontos Interiores convencional utiliza uma percentagem fixa de combinação entre as direções de Centralização e Afim-Escala. Apesar das suas boas características de convergência nos problemas práticos, o valor de combinação utilizado, isto é, aquele que afeta diretamente o número de iterações do processo iterativo, é empírico, não sendo portanto ótimo;
- Neste trabalho, analisou-se as direções que compõem a trajetória dos métodos de Pontos Interiores e sua combinação numa direção de busca, tendo-se observado que a referida combinação das direções pode ser feita segundo vários critérios;
- Algumas versões melhoram a velocidade da convergência, reduzindo o número de iterações (como os algoritmos Preditor-Corretor e Preditor-Corretor com Múltiplas Correções). Estes algoritmos podem ser apropriados quando se requer reduzido esforço computacional em problemas com grau de complexidade moderado;
- Outras versões procuram direções centralizadas para atingir o ponto candidato a ótimo (como os algoritmos de Pontos Interiores Centralizado, Direções Sucessivas Fortemente Centralizado e Máximo Passo). Estes algoritmos podem ser utilizados em problemas com regiões de soluções viáveis muito restritas;
- Algumas das versões estudadas são úteis para a análise das características de convergência e da combinação das direções do método de Pontos Interiores (como os algoritmos de Direções Sucessivas e Otimização na Direção de Busca);
- A maioria dos algoritmos lineares testados mostrou-se adequada para a solução do problema não-linear de FPO;

- A utilização adequada de Processamento de Alto Desempenho pode melhorar a performance computacional dos métodos de Pontos Interiores. A metodologia proposta de vetorização dos algoritmos via "caminho" de fatoração mostrou-se eficiente, resultando em ganhos significativos na resolução dos sistemas lineares;
- Os testes realizados com sistemas realísticos de médio e grande porte mostraram o potencial dos métodos analisados na solução de problemas de otimização não-lineares e de grande porte.

Finalmente, os métodos de Pontos Interiores, apesar de surgirem de um algoritmo comum, formam hoje uma família de algoritmos. A seleção e adequação da versão a ser utilizada numa aplicação particularizada constitui um problema de engenharia que redundará na eficácia do algoritmo.

7.3. Sugestões para Futuros Trabalhos

Algumas sugestões para futuros trabalhos são:

- A pesquisa na área matemática dos métodos de Trajetória Central encontra-se em grande desenvolvimento. Um adequado acompanhamento desta pesquisa pode fornecer ao engenheiro novas ferramentas que melhorem a performance da resolução dos problemas práticos que enfrenta;
- O estudo das características práticas das versões do método de Pontos Interiores deve ser aprofundado, para permitir a sua aplicação em problemas reais;
- A vetorização dos algoritmos apresentados poderia considerar a utilização das capacidades de processamento paralelo dos computadores vetoriais;
- Dentro da área da pesquisa matemática, sugere-se a consideração, além das direções de Centralização e Afim-Escala, da combinação de duas outras direções: a de Viabilidade das Restrições de Igualdade e o Gradiente da Função Objetivo.

Bibliografia

- ALMEIDA, K.C.; GALIANA, F.D.; 1996. Critical Cases in the Optimal Power Flow. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v.11, n. 3, Aug. 1996, p. 1509-1518.
- ANDRUS, J.F.; 1992. An Exterior Point Method for the Convex Programming Problem. *Journal of Optimization Theory and Applications*. USA, v. 72, n.1, Jan. 1992. p. 37-63
- ANDRUS, J.F.; SCHAFERKOTTER, M.R.; 1996. An Exterior-Point Method for Linear Programming Problems. *Journal of Optimization Theory and Applications*. USA, v. 91, n.3, Dez. 1996. p. 561-583
- AOKI, M; 1971. *Introduction to Optimization Techniques - Nonlinear Programming*. New York, USA: Macmillan Company.
- BARBEAU, E.J.; 1989. *Polynomials*. New York, USA, Springer-Verlag New York Inc., 455p.
- BARBOZA, Luciano Vitoria; 1997. *Análise do Máximo Carregamento de Sistemas de Potência via Métodos de Pontos Interiores*. Florianópolis, 1997. Dissertação (Mestrado em Eng. Elétrica) – Centro Tecnológico, Universidade Federal de Santa Catarina.
- BERTSEKAS, Dimitri P.; 1996. *Constrained Optimization and Lagrange Multiplier Methods*. USA: Academic Press, Inc.
- BURCHET, R.C.; HAPP, H.H.; NIERATH, D.R.; 1984. Quadratically Convergent Optimal Power Flow. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v. 103 n.11 (Nov.).
- CARAMANIS, M.C.; BOHN, R.E.; SCHWEPPE, F.C.; 1982. Optimal Spot Pricing: Practice and Theory. *IEEE Transactions on Power Apparatus and Systems*, v. PAS-101, n. 9, (Sept.) 1982.
- CARPENTIER, J.; 1962. Contribution à l'étude du dispatching économique. *Bulletin de la Société Française des Electriciens*, ser.8, vol.3, pp. 431-447, 1962.
- CARPENTIER, J; 1972. Results and Extensions of the Methods of Differential Injections. In: POWER SYSTEMS COMPUTATION CONFERENCE (4:1972). *Proceedings*. 1972. v2, 2.1/8.

- CARVALHO, Ricardo José de Oliveira; SALGADO, Roberto; 1997. Fluxo de Potência Ótimo Multi-Objetivo: Abordagem via Pontos Interiores. In: CONGRESSO CHILENO DE INGENIERIA ELECTRICA (12.:1997:Temuco). *Anais*. Temuco, Chile, 1997.
- CARVALHO, Ricardo José de Oliveira; 1999. *Fluxo de Potência Ótimo: Algoritmos de Pontos Interiores, Abordagem Multi-Objetivo e Aplicação de Transformações Ortogonais*. Florianópolis, 1999. Tese (Doutorado em Eng. Elétrica) – Centro Tecnológico, Universidade Federal de Santa Catarina.
- CASTRONUOVO, E.D.; CAMPAGNOLO, J.M.; SALGADO, R.; 1998a. Fluxo de Potência Ótimo via Método de Pontos Interiores Não-Linear Primal-Dual, utilizando Computação de Alto Desempenho. In: SBAC-PAD, SEMINÁRIO BRASILEIRO DE ARQUITETURA DE COMPUTADORES E PROGRAMAÇÃO DE ALTO DESEMPENHO. (13: set. 1998: Buzios, RJ.) *Anais*. Rio de Janeiro, 1998.
- CASTRONUOVO, E.D.; CAMPAGNOLO, J.M.; SALGADO, R.; 1998b. Vetorização do Método de Pontos Interiores Não-Linear Primal-Dual e sua Aplicação ao Fluxo de Potência Ótimo. In: CBA, CONGRESSO BRASILEIRO DE AUTOMÁTICA. (12 :set. 1998: Uberlândia, MG). *Anais*. Uberlândia, 1998.
- CASTRONUOVO, E.D.; CAMPAGNOLO, J.M.; SALGADO, R.; 1999. Optimal Power Flow Solutions via Interior Point Method with High Performance Computation Techniques. In: PSCC'99, 13th Power Systems Computation Conference. (Jun: 1999: Trondheim, Norway. *Proceedings*.
- CASTRONUOVO, E.D.; CAMPAGNOLO, J.M.; SALGADO, R.; 2000a. On the Applications of High Performance Computation Techniques to Nonlinear Interior Point Methods. *IEEE Transactions on Power Systems*, aceito para publicação.
- CASTRONUOVO, E.D.; CAMPAGNOLO, J.M.; SALGADO, R.; 2000b. A Largest-Step Central-Path Algorithm Applied to the Optimal Power Flow Problem. *Journal de Automação e Control*. Campinas, SP, Brasil, v. 11, Dez. 2000.
- CASTRONUOVO, E.D.; CAMPAGNOLO, J.M.; SALGADO, R.; 2000c. Processamento Vetorial do Algoritmo Não-Linear Preditor-Corretor Primal-Dual de Pontos Interiores. In: XIII CBA, Congresso Brasileiro de Automática. (11,14: set. : 2000: Florianópolis, SC: Brasil). *Anais*.

- CASTRONUOVO, Edgardo Daniel; 1997. *Solução do Fluxo de Potência Ótimo via Método de Pontos Interiores, com Aplicação de Computação de Alto Desempenho*. Florianópolis, 1997. Dissertação (Mestrado em Eng. Elétrica) – Centro Tecnológico, Universidade Federal de Santa Catarina.
- CHAIR, James D. McCalley; ASGARPOOR, Sohrab; GEDRA, Thomas; HALPIN, Mark; SAINI, Narinder K.; SCHRAMEYER, Michael; 1997. Second Bibliography on Transmission Access Issues. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v.12, n. 4, (nov.) p. 1654-1659.
- CHENG, J.W.M.; GALIANA, F.D.; MCGILLIS, D.T.; 1997. Studies of Bilateral Contracts with respect to Steady-State Security in a Deregulated Environment. Power Systems Computer Application Conference, Denver USA. (May.).
- COVA, B.; GRANELLI, G.P.; MONTAGNA, M.; SILVESTRI, A.; INNORTA, M.; MARANINNO, P.; 1987. Large Scale Application of the Han-Powell Algorithm to Compact Models of Static and Dynamic Dispatch of Real Power. *Int. Journal of Electrical Power and Energy Systems*, v. 9, n. 3, p. 130-141.
- CRAY Instruction Manual. Version A. Mendota Heigts: CRAY Research, Inc., 1994.
- DANDACHI, N.H.; RAWLINS, M.J.; ALSAÇ, O.; PRAIS, M.; STOTT, B. ; 1997. OPF for Reactive Pricing Studies on the NGC System. Power Systems Computer Application Conference, Denver USA. (May.).
- DANTZIG, G.B.; 1963. *Linear Programming and Extensions*. Princeton, NJ, USA: Princeton University Press.
- DECKER, I.C.; CASTRO, M.C.A.; VANTI, M.R.V.;1998. Melhoria da Segurança Dinâmica de Sistemas de Energia Elétrica Utilizando Processamento Distribuído. XIII SBAC-PAD, Seminário Brasileiro de Arquitetura de Computadores e Programação de Alto Desempenho. Buzios, RJ, (set.).
- DOMMEL, H.W.; TINNEY, W.; 1968. Optimal Power Flow Solutions. *IEEE Transactions on Power Apparatus and Systems*, Piscataway, NJ, USA , PAS 87, n. 10, (Oct.).
- DONGARRA, J.J., DUFF, I.S., SORENSEN, D.C., van der VORST, H.A. *Numerical Linear Algebra for High-Performance Computers*. Philadelphia, PA, USA: SIAM. 342p.

- DY LIACCO, T.E.; 1974. Real-Time Computer Control of Power Systems. Proceedings of the IEEE, v. 62, n. 7, (July).
- DY LIACCO, T.E.; 1978. System Security: the Computer' Role. *IEEE Spectrum*, (Jun).
- EL-BAKRY, A.S.; TAPIA, R.A.; TSUCHIDA, T.; ZHANG, Y; 1996. On the Formulation and Theory of the Newton Interior-Point Method for Nonlinear Programming. *Journal of Optimization Theory and Applications*. USA, v. 89, n. 3, June 1996.
- EL-HAWARY, M.E.; 1996. Optimal Power Flow: Solution Techniques, Requirements and Challenges. In: *Optimal Power Flow: Solution Techniques, Requirements and Challenges, IEEE Tutorial Course*. New York: The Institute of Electrical and Electronics Engineers, Inc., 1996. v-v
- ENS, M.K., TINNEY, W.F., ALVARADO, F.L.; 1990. Sparse Matrix Inverse Factors. *IEEE Transactions in Power Systems*, Piscataway, NJ, USA, v. 5. n. 2, May 1990, p. 466-473.
- FIACCO, A.V.; McCORMICK, G.P.; 1968. *Nonlinear Programming*. New York, USA: John Wiley.
- FLETCHER, R.; 1987. *Practical Methods of Optimization*. 2th. Edition. Reino Unido: John Wiley & Sons.
- FLUECK, A.J.; CHIANG, H.D.; 1998. Solving the Nonlinear Power Flow Equations with an Inexact Newton Method using GMRES. *IEEE Transactions in Power Systems*, Piscataway, NJ, USA, v. 13. n. 2, May 1998, p. 267-273.
- FORREST, J.J.H., TOMLIN, J.A. Vector Processing in Simplex and Interior Methods for Linear Programming. *Annals of Operations Research*, 22(1990), pág. 71-100.
- FORSREN, A.; GILL, Philip E.; 1996. *Primal-Dual Interior Methods for Nonconvex Nonlinear Programming*. San Diego, CA, USA, 1996. Reporte Técnico NA 96-3. Department of Mathematical, University of California.
- FRITZSCHE, Helmut; 1978. *Programação Não-Linear: análise e métodos*. São Paulo: Ed. Da Universidade de São Paulo.
- GALIANA, F.D.; JAVIDI, H.; McFEE, S.; 1994. On the Application of a Preconditioned Conjugate Gradient Algorithm to Power Network Analysis. *IEEE Transactions in Power Systems*, Piscataway, NJ, USA, v. 9, n. 2, May 1994, p. 625-636.

- GALIANA, F.D.; ILIC, M.; 1998. Mathematical Framework to the Analysis and Management of Power Transactions under Open Access - *IEEE Transactions on Power Systems*, Vol 13, N°2, May
- GIRAS, T.C.; TALUKDAR, S.N.; 1981. Quasi-Newton Method for Optimal Power Flows. *Int. Journal of Electrical Power and Energy Systems*, v. 3, n. 2, p. 59-64.
- GOMEZ, A., BETANCOURT, R.; 1990. Implementation of the Fast Decoupled Load Flow on a Vector Computer. *IEEE Transaction on Power Systems*, Piscataway, NJ, USA, v. 5, n. 3, pág. 977-983.
- GOMEZ, A.; FRANQUELO, L.G.; 1988. Node Ordering Algorithms for Sparse Vector Method Improvement. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v. 3, n. 1, (Feb.).
- GONZAGA, C.C.; BONNANS, J.F.; 1996. Fast Convergence of the Simplified Largest Step Path Following Algorithm. *Mathematical Programming*. USA, v.76, 1996. p. 95-115
- GRANVILLE, Sergio; 1994. Optimal Reactive Dispatch through Interior Point Methods. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v. 9, n. 1, (Feb.).
- GRANVILLE, S.; MELLO, J. C. O.; MELO, A. C. G.; 1996. Application of Interior Point Methods to Power Flow Unsolvability. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v.11, n. 2, (May.), p. 1096-1103.
- GRIBIK, Paul R.; SHIRMOHAMMADI, Dariush; HAO, Shagyuo; THOMAS, Chifong L.; 1990. Optimal Power Flow Sensitivity In: IEEE/PES WINTER MEETING (1990). *Proceedings*. Atlanta, Georgia, USA.
- GUAN, X.; LIU, E.; PAPALEXOPOULOS, A.D.; 1995. Application of a Fuzzi Set Method in an Optimal Power Flow. *Electric Power System Research*, v. 34, n. 2, 1995.
- HESTENES, M.R.; 1969. Multiplier and Gradient Methods. *Journal of Optimization Theory and Applications*. v. 4, p. 303-329.
- HUANG, H.S.; LU, C.N.; 1993. Efficient Storage Scheme and Algorithms for W-Matrix Vector Multiplication on Vector Computers. IEEE 93SM538, Vancouver, Summer Meeting.
- HUNEULT, M.; GALIANA, F.D.; 1991. A Survey of the Optimal Power Flow Literature. *IEEE Transaction on Power Systems*, Piscataway, NJ, USA, v. 6, n. 2, May 1991.

- ILIC, M.; GALIANA, F.; FINK, L.; 1998. Power Systems Restructuring: Engineering and Economics. Kluwer Academic Publishers. Norwell, USA, p. 559.
- IRRISARI, G.D.; WANG, X.; TONG, J.; MOKHTARI, S.; 1996. Maximum Loadability of Power Systems Using Interior Point Non-Linear Optimization Method. In: IEEE/PES WINTER MEETING (1996). *Proceedings*. 1996.
- KARMARKAR, N.; 1984. A New Polynomial-Time Algorithm for Linear Programming. *Combinatorica*, USA, v.4, 1984. p. 373-395
- KUHN, H.W.; TUCKER, A.W.; 1951. Nonlinear Programming. In: BERKELEY SYMPOSIUM ON MATHEMATICAL STATISTICS AND PROBABILITY (2: 1951: Berkeley, USA) *Proceedings*. Berkeley, USA, 1951.
- LANKFORD, C. B.; McCALLEY, J. D.; SAINI, N.K.; 1996 Bibliography on Transmission Access Issues. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v. 11, n. 1, (Feb.), p. 30-40.
- LEWIS, T.G. ;1993. Foundations of Parallel Programming: A Machine-Independent Approach. IEEE Computer Society Press. Los Alamitos, California, USA.
- LUENBERGER, David G.; 1965. *Introduction to Linear and Nonlinear Programming*. USA: Addison-Wesley Publishing Company, Inc.
- MEHROTRA, Sanjay; 1992. On The Implementation of a Primal-Dual Interior Point Method. *SIAM J. Optimization*. v. 2, n. 4, (Nov.), p. 576-601.
- MERLIN, a.; 1972. On the Optimal Generation Planning in the Large Transmission System (The MAYA Model). In: POWER SYSTEMS COMPUTATION CONFERENCE (4:1972). *Proceedings*. 1972. v2, 2.1/6.
- MOMOH, J.A.; 1996. Interior Point Methods and Variants for OPF. In: *Optimal Power Flow: Solution Techniques, Requirements and Challenges, IEEE Tutorial Course*. New York: The Institute of Electrical and Electronics Engineers, Inc., 1996. p. 26-35
- MONTAGNA, M.; GRANELLI, G.P.; VUONG, G.T. ;1996. Levelwise Algorithms for Vector Processing of Sparse Power Systems Matrices. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA. v. 11, n. 1, (Feb.).

- MONTAGNA, M.; GRANELLI, G.P.; VUONG, G.T.; MARANNINO, P.; 1992. Vector Computer Implementation of Power Flow Outage Studies. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA , v. 7, n. 2, (May.).
- MONTICELLI, A.; LIU, W.H.E.; 1992. Adaptive Movement Penalty Method for the Newton Optimal Power Flow. *IEEE Transactions in Power Systems*, Piscataway, NJ, USA, v. 7, n. 1, Feb 1992, p. 334-342.
- MORI, H.; TANAKA, H.; KANNO, J; 1996. A Preconditioned Fast Decoupled Power Flow Method for Contingency Screening. *IEEE Transactions in Power Systems*, Piscataway, NJ, USA, v. 11, n. 1, Feb. 1996, p. 357-363.
- MUKERJI, R.; NEUGEBAUER, W; LUDORF, R.P.; CATELLI, A.; 1992. Evaluation of Wheeling and Non-Utility Generation (NUG) Options using Optimal Power Flows. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v. 7, n. 1 (Feb.), p. 201-207.
- NABONA, N.; FERRIS, L.L.; 1973. Optimization of Economic Dispatch through Quadratic and Linear Programming. *Proc. of IEE*, v. 120, n. 5, p. 574-580.
- NICHOLSON, H.; STERLING, M.J.H.; 1973. Optimum Dispatch of Active and Reactive Generation by Quadratic Programming. *IEEE Transactions in Power Apparatus e Systems*, Piscataway, NJ, USA, v.PAS-92, 1984, p.644-654.
- PADILLA, A., MORELATO, A.; 1992. A W-matrix Methodology for Solving Sparse Network Equations on Multiprocessor Computers. *IEEE Transactions in Power Systems*, Piscataway, NJ, USA, v. 7, n. 2, May 1990, p. 452-459.
- PAPALEXOPOULOS, A.D.; IMPARATO, C.F.; WU, F.F.; 1989. Large-Scale Optimal Power Flow: Effects of Initialization, Decoupling and Discretization. *IEEE Transactions in Power Systems*, Piscataway, NJ, USA, v. 4, n. 2, May 1989, p. 748-759.
- PAPALEXOPOULOS, Alex D.; 1996. Challenges to On-Line OPF Implementation. In: *Optimal Power Flow: Solution Techniques, Requirements and Challenges, IEEE Tutorial Course*. New York:The Institute of Electrical and Electronics Engineers, Inc., 1996, p. 36-51.
- PARKER, C.J.; MORRISON, I.F.; SUTANTO, D.; 1995. Application of an Optimisation Method for Determining the Reactive Margin from Voltage Collapse in Reactive Power Planning. In: *IEEE/PES SUMMER MEETING (July, 1995:Portland). Proceedings*. Portland, Oregon, USA, 1995.

- QUINN, Michael J.; 1987. *Parallel Computing, Theory and Practice*. 2nd. Ed. New York, USA: McGraw-Hill, p. 445.
- RAJARAMAN, Rajesh; SALLASHKAR, Jayant; ALVARADO, Fernando; 1997. The effect of Demand Elasticity on Security Prices for The Poolco and Multi-Lateral Contract Models. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v.12, n. 3, (Aug.), p. 1177-1184.
- RAMANATHAN, R.; 1996. Optimal Power Flow Applications and Training Requirements. In: *Optimal Power Flow: Solution Techniques, Requirements and Challenges, IEEE Tutorial Course*. New York: The Institute of Electrical and Electronics Engineers, Inc., 1996, p. 52-60.
- RENEGAR, J.; 1988. A Polynomial Time Algorithm, based on Newton's Method, for Linear Programming. *Mathematical Programming*. USA, v.40, 1988. p. 59-93
- RISTANOVIC, Petar; 1996. Successive Linear Programming Based OPF Solution. In: *Optimal Power Flow: Solution Techniques, Requirements and Challenges, IEEE Tutorial Course*. New York: The Institute of Electrical and Electronics Engineers, Inc., 1996. 1-9.
- RISTANOVIC, Petar; 1996. Successive Linear Programming Based OPF Solution. In: IEEE/PES WINTER MEETING (1996: Baltimore). *Proceedings*. Baltimore, USA, 1996.
- SALGADO, Roberto; 1999. *Otimização Estática Aplicada a Sistemas de Potência, Notas de Aula*. Curso de Pós-Graduação em Eng. Elétrica. UFSC. Florianópolis, 1999
- SANCHEZ, C.A.; 1983. *Formulação Matemática de um Método de Minimizações Seqüenciais sem Restrições para Pontos Interiores*. Dissertação de Mestrado. Departamento da Matemática. UFSC. Florianópolis.
- SHAHIDEHPOUR, S.M.; RAMESH, V.C.; 1996. Nonlinear Programming Algorithms and Decomposition Strategies for OPF. *IEEE Tutorial Course of Optimal Power Flow*.
- SHANNO, D.F.; VANDERBEI, R.J.; 1999. *Interior-Point Methods for Nonconvex Nonlinear Programming: Orderings and Higher-Order Methods*. Princeton University. Reporte Técnico SOR-99-5.
- SHEBLE, G.B.; 1990. Optimization Overview with Economic Dispatch Examples. In: *Application of Optimization Methods for Economy/Security Functions in Power System*

Operations, IEEE Tutorial Course. New York: The Institute of Electrical and Electronics Engineers, Inc., 1990. 1-13

SHIRMOHAMMADI, D.; RAJAGOPALAN, C.; ALWARD, E.R.; THOMAS, C.L.; 1991. Cost of Transmission Transactions: an Introduction. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v. 6, n. 4, (Nov.)

SOMAN, S.A.; PARTHASARATHY, K.; THUKARAM, D.; 1994. Curtailed Number and Reduced Controller Movement Optimization Algorithms for Real Time Voltage/Reactive Power Control. *IEEE Transaction on Power Systems*, Piscataway, NJ, USA, v. 9, n. 4, Nov. 1994.

STERN, Julio M.; 1994. *Esparsidade, Estrutura, Estabilidade e Escalonamento em Álgebra Linear Computacional*. Recife, UFPE-DI, p.123.

STOTT, B.; ALSAC, O.; 1974. Fast Decoupled Load Flows. *IEEE Transaction on Power Apparatus and Systems*, Piscataway, NJ, USA, v. PAS-93, May/June 1974, p. 859-869.

STOTT, B.; HOBSON, E.; 1977a. Power System Security Control Calculations using Linear programming, Part 1. In: IEEE/PES SUMMER MEETING (July 1977). *Proceedings*. Ciudad de México, México.

STOTT, B.; MARINHO, J.L.; 1979. Linear Programming for Power System Network Security Applications. *IEEE Transaction on Power Apparatus and Systems*, Piscataway, NJ, USA, v. 98, n. 3, 1979. p. 837-848.

SUN, D.I.; ASHLEY, B.; BREWER, B.; HUGHES, A.; TINNEY, W.F.; 1984. Optimal Power Flow by Newton Approach. *IEEE Transactions on Power Apparatus and Systems*, v. PAS-103, n. 10. (Oct.)

SUN, J.; ZHAO, G.Y.; 1998. Quadratic Convergence of a Long-Step Interior-Point Method for Nonlinear Monotone Variational Inequality Problems. *Journal of Optimization Theory and Applications*. v. 9, n. 2, (May).

TALUKDAR, S.N.; GIRAS, T.C.; 1981. A Fast and Robust Variable Metric Method for Optimal Power Flows. *IEEE Transactions in Power Apparatus e Systems*, Piscataway, NJ, USA, v.101. n. 2, 1981, p. 415-420.

- TAPIA, R.; ZHANG, Y.; SALTZMAN, M.; WEISER, A.; 1996. The Mehrotra Predictor-Corrector Interior-Point Method as a Perturbed Composite Newton Method. *SIAM J. Optimization*, USA, v. 6, n.1 (Feb.), p. 47-56.
- TINNEY, W.F.; BRIGHT, J.M.; DEMAREE, K.D.; HUGHES, B.A.; 1988. Some Deficiencies in Optimal Power Flow. *IEEE Transaction on Power Systems*, Piscataway, NJ, USA, v. 3, n. 2, May 1988.
- VANDERBEI, R.J.; SHANNO, D.F.; 1997. *An Interior-Point Algorithm for Nonconvex Nonlinear Programming*. Princeton University. Reporte Técnico SOR-97-21.
- VARGAS, L.S.; QUINTANA, V.H.; VANNELLI, A.; 1993. A Tutorial Description of an Interior Point Method and its applications to Security-Constrained Economic Dispatch. *IEEE Transaction on Power Systems*, Piscataway, NJ, USA, v. 8, n. 3, Aug. 1993.
- VILLELA PINTO, M. A.; 1997. *Um Estudo Computacional de Algoritmos de Trajetória Central para Problemas de Complementaridade Linear Monotônica*. Dissertação de Mestrado em Matemática e Computação Científica. Universidade Federal de Santa Catarina. Florianópolis, SC, (dez.).
- WANG, Tao; MONTEIRO, R.D.C.; PANG, J.S.; 1995. *An Interior Point Potential Reduction Method for Constrained Equations*. Baltimore, Maryland, USA, 1995. Reporte Técnico. Department of Mathematical Sciences, The Johns Hopkins University.
- WEI, Hua; SASAKI, H.; KUBOKAWA, J.; YOKOYAMA R.; 1998. An Interior Point Nonlinear Programming for Optimal Power Flow Problems with A Nobel Data Structure. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v.13, n. 3, (Aug.), p. 870-877.
- WRIGHT, Stephen J.; 1997. *Primal-Dual Interior Point Methods*. Philadelphia, PA, USA:Society for Industrial and Applied Mathematics (SIAM).
- WU, Yu-Chi; DEBS, Atif S.; MARSTEN, Roy E. ;1994. A Direct Nonlinear Predictor-Corrector Primal-Dual Interior Point Algorithm for Optimal Power Flows. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v.9, n. 2, (May), p.876-883.
- YOKOYAMA, R; BAE, S.H.; MORITA, T.; SASAKI, H.; 1988. Multiobjective Optimal Generation Dispatch Based on Probability Security Criteria. *IEEE Transaction on Power Systems*, Piscataway, NJ, USA, v. 3, Feb. 1988.

- YU; NESTEROV; 1996. Long-Step Strategies in Interior-Point Primal-Dual Methods. *Mathematical Programming*. USA, v.76, 1996. p. 47-94
- ZANGWILL, W.; 1969. *Nonlinear Programming: a Unified Approach*. New York, USA: Prentice-Hall, Inc.
- ZHANG, Y.; TINNEY, W.F.; 1994. Partial Refactorization with Unrestricted Topology Changes. *IEEE/PES 1994 Summer Meeting*. San Francisco, USA, (July) p. 24-28.
- ZHANG, Yin; ZHANG; 1995 On Polynomiality of the Mehrotra-type predictor- corrector interior-point algorithms. In: *Mathematical programming*. USA, v.68, p. 303-318.
- ZOBIAN, A.; ILIC, M.D.; 1997a. Unbundling of Transmission and Ancillary Services. Part I: Technical Issues. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v.12, n. 2, (May).
- ZOBIAN, A.; ILIC, M.D.; 1997b. Unbundling of Transmission and Ancillary Services. Part II: Cost-Based Pricing Framework. *IEEE Transactions on Power Systems*, Piscataway, NJ, USA, v. 12, n. 2, (May).
- ZOLLENKOPF, K.; 1987. Basic Computation Algorithm and Programming Techniques. In: *Large Sparse Sets of Linear Equations*. New York, USA: Academic Press, pág. 75-97