

UNIVERSIDADE FEDERAL DE MINAS GERAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA
CENTRO DE PESQUISA E DESENVOLVIMENTO EM ENGENHARIA ELÉTRICA

Projeto de Redes de Distribuição de Energia com Incertezas na Evolução de Carga Utilizando Algoritmos Meméticos

Bruno Baêta e Souza

Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais, como requisito parcial para obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Prof. Oriane Magela Neto

Co-orientador: Prof. Ricardo Hiroshi Caldeira Takahashi

Belo Horizonte, Dezembro de 2007

Resumo

Redes de distribuição de energia elétrica têm uma importância econômica e social muito destacada uma vez que elas possibilitam o fornecimento de energia elétrica para cerca de 85% da população do país. Esta inquestionável importância, associada ao alto volume de dinheiro requerido para a instalação desses sistemas, justificam amplamente a aplicação de técnicas de otimização no planejamento dessas redes. Sistemas de distribuição eficientes tendem a fazer melhor uso dos recursos disponíveis, reduzindo os desperdícios de energia e os custos necessários à sua instalação. O problema de projeto ótimo de redes é de natureza altamente complexa, devido à característica discreta do espaço onde as soluções são definidas e aos funcionais não-lineares inerentes ao problema. Essas razões tornam inviável o emprego de técnicas tradicionais de otimização, como métodos determinísticos contínuos. Uma classe de algoritmos de otimização que vem sendo utilizada na otimização dessas redes são os algoritmos evolucionários, devido às suas flexibilidade, robustez e capacidade de adaptação. Os Algoritmos Meméticos, que são a hibridação entre um algoritmo evolucionário e um método de busca local, também têm se tornado uma alternativa importante de otimização de redes. Eles são ferramentas reconhecidamente poderosas para a obtenção e refinamento de ótimos locais e globais. Os algoritmos meméticos desenvolvidos neste trabalho são decorrentes da hibridação entre o Algoritmo de Seleção Clonal, que é capaz de retornar como resposta do problema de otimização um conjunto de soluções formado pela solução ótima e por soluções sub-ótimas e métodos de busca local desenvolvidos a partir da utilização da métrica T -norm, que propicia a geração aleatória de redes à distâncias pré-definidas. Com a obtenção do conjunto de soluções ótimas por meio dos algoritmos meméticos, é

possível tratar uma questão muito importante do projeto de redes que são as incertezas associadas à evolução da carga do sistema. As incertezas são tratadas por meio de uma análise de sensibilidade multiobjetivo, onde cenários diversos são obtidos por Simulação de Monte Carlo e cada uma das soluções do conjunto são avaliadas segundo quatro critérios considerados relevantes. Uma vez obtido o conjunto de soluções não-dominadas pela análise de sensibilidade, torna-se possível ao projetista escolher qual rede melhor se adapta aos requisitos específicos do projeto, levando em consideração que a rede escolhida deve ser robusta quanto a possíveis variações nas condições de carga. Com a aplicação dos algoritmos meméticos desenvolvidos neste trabalho foi possível obter um conjunto de redes bem robustas às incertezas do projeto, provendo o projetista de um conjunto de alternativas viáveis. Um resultado importante é a possibilidade de se obter redes altamente robustas com um pequeno aumento no capital que seria investido na instalação de uma rede projetada para o cenário médio de variação da carga.

Abstract

Electric power distribution systems have highlighted economic and social importance since they are responsible for supplying 85% of the country population. This unquestionable importance, together with the high budget usually required for installing those systems, justify the use of optimisation techniques in the distribution system design. Optimised distribution networks provide better allocation of the financial resources, reducing total installation costs and power losses. The distribution network design is intrinsically complex due to the discrete characteristic of the search space and the nonlinearities involved. Therefore the traditional optimisation algorithms such as deterministic continuous methods become unviable. An optimisation algorithm class that has been recently used in the optimisation of such systems is the evolutionary algorithms due to their flexibility, robustness and capacity of adaptation. Memetic Algorithms, which are a hybridization from an evolutionary algorithm and a local search method, emerge as an interesting option since they are a powerful tool for obtaining and refining global and local optima. Memetic algorithms proposed here are the hybridization from a Clonal Selection Algorithm, which is able to find a set of optimal and sub-optima solutions, and a local search method, which is based on the the *T-norm* network metric, that generates random networks at pre-defined distances. The set of optimal solutions mapped by the memetic algorithms are very useful for dealing with an important aspect of the electric distribution network design - the load evolution uncertainties. They are treated by means of a multiobjective sensitivity analysis. A Monte Carlo Simulation is used to provide random possible load scenarios. The set of non-dominated solutions found by the sensitivity analysis helps the designer in his choice of the best network.

Possible changes in load conditions are taken into account and robust networks, which can handle with design uncertainties, are obtained. An important result gained here is the possibility of using a highly robust networks spending just a few money more than that which would be used for the construction of a network designed for the medium load scenario.

Agradecimentos

Agradeço a minha família por todas as conquistas, pelo apoio, compreensão, exemplo recebido em toda minha vida. Em especial, agradeço a meu pai e minha mãe e dedico este trabalho a eles. Minhas irmãs, Priscilla, Dindinha e Éric agradeço por sempre estarem ao meu lado e me apoiando.

Tio Mauri e tia Glória, um agradecimento especial a vocês e sua família, que me acolheram com tanto carinho. Foi uma alegria imensa ser recebido em sua casa durante toda esta batalha. Douglas, Diego, Deividson, valeu demais!!!!

Agora um agradecimento coletivo, já que a família é enorme, aos meus tios e suas famílias: Zé Antônio, Elias, Geraldo, Carlos, Estevão, Aloísio, Edson, Lúcio, Adão, Mauri, Carlos, Dindinha, Odilon, Oscar, Dioguinho e Flavinho, muito obrigado por tudo!

Aos professores Oriane e Takahashi pela orientação, pelo conhecimento adquirido e principalmente pela amizade construída ao longo deste período. Obrigado também professor Rodney, sempre disposto a ensinar e com idéias muito interessantes. E ao CPDEE, professores e funcionários, pela amizade e cooperação durante este período.

Eduardo Carrano e Gledson, valeu muito a ajuda de vocês. Gledson sempre dando aquela força no Latex e Eduardo além de sempre ajudando nos algoritmos é um amigão desde muito tempo, antes mesmo da república em São João. Aproveitando, obrigado também aos amigos do CPDEE, do GOPAC, LABSIS, UFSJ um abraço pra vocês.

Agradeço também à CAPES pelo apoio financeiro.

Sumário

Lista de Figuras	viii
Lista de Tabelas	x
1 Projeto de Redes de Distribuição de Energia Elétrica	1
1.1 Introdução ao Projeto de Redes de Distribuição de Energia Elétrica . .	1
1.2 Representação das Variáveis do Projeto de Redes de Distribuição de Energia Elétrica	5
1.2.1 Grafos	5
1.2.2 Representação das Variáveis	7
1.3 Controle Dimensional em Problemas de Redes	9
1.3.1 <i>Controlled-Greedy Encoding</i>	10
1.4 Representação das Redes no Espaço Contínuo	12
1.4.1 O Conceito de Distância no Espaço de Redes - <i>Métrica T-norm</i>	14
1.4.2 Geração de Redes Aleatórias à Distâncias Pré-definidas	17
1.5 Formulação do Problema de Redes de Distribuição de Energia Elétrica	20
2 Algoritmos de Otimização	24
2.1 Descrição do Algoritmo de Seleção Clonal	24
2.1.1 Inspiração Biológica	24
2.1.2 Estrutura Básica do Algoritmo de Seleção Clonal	33
2.2 Algoritmos Meméticos	35
2.2.1 Descrição	35
3 Algoritmos Utilizados no Projeto de Redes de Distribuição de Energia Elétrica	38
3.1 Algoritmo Clonal para o Projeto de Redes de Distribuição de Energia Elétrica	38
3.2 Definição de Vizinhança	43
3.3 Método de Busca Local	46
3.4 Algoritmos Meméticos Adaptados ao Projeto de Redes de Distribuição	50
3.5 Análise de Sensibilidade Multiobjetivo das Soluções	53
3.5.1 Modelo de Crescimento da Carga	56

3.5.2	Modelo de Variação do Preço da Energia	56
3.5.3	Distribuição de Probabilidade Conjunta das Incertezas	57
3.5.4	Simulação de Monte Carlo	57
4	Resultados	59
4.1	O Sistema de 21 nós	61
4.1.1	Resultados Obtidos via Algoritmo Clonal	62
4.1.2	Resultados Obtidos via AM1	65
4.1.3	Resultados Obtidos via AM2	73
4.1.4	Resultados Obtidos via AM3	78
4.1.5	Comparação dos Resultados Obtidos para o Caso de 21 nós	82
4.2	O Sistema de 100 nós	86
4.2.1	Resultados Obtidos via Algoritmo Clonal	87
4.2.2	Resultados Obtidos via AM3	91
4.2.3	Comparação dos Resultados Obtidos para o Caso de 100 nós	91
5	Conclusões	96
6	Propostas de Continuidade	98
	Referências Bibliográficas	100
A	Redes obtidas pelas simulações dos algoritmos de otimização	106
A.1	Caso de 21 nós	107
A.1.1	Primeira simulação com o algoritmo clonal	107
A.1.2	Segunda simulação com o algoritmo clonal	108
A.1.3	Primeira simulação com o AM1	109
A.1.4	Segunda simulação com o AM1	110
A.1.5	Primeira simulação com o AM2	111
A.1.6	Segunda simulação com o AM2	112
A.1.7	Primeira simulação com o AM3	113
A.1.8	Segunda simulação com o AM3	114
A.2	Caso de 100 nós	115
A.2.1	Algoritmo clonal	115
A.2.2	AM3	116

Lista de Figuras

1.1	Exemplo de grafo	6
1.2	Exemplo de grafo completo	6
1.3	Exemplo de árvore	8
1.4	Número de conexões <i>vs.</i> distância média	11
1.5	Redes exemplos para a métrica <i>T-norm</i>	18
2.1	Mecanismos de defesa e seus principais mediadores.	26
2.2	Resposta imunológica primária, secundária e reativa cruzada	31
3.1	Exemplo de rede para calcular RL	44
3.2	Modelo de crescimento de carga	58
4.1	Sistema de 21 nós - configuração inicial	62
4.2	Sistema de 21 nós - codificação pela <i>Controlled-Greedy Encoding</i>	62
4.3	Rede ótima para o cenário mais provável	64
4.4	Redes obtidas pela otimização via Algoritmo Clonal - Primeira Simulação.	65
4.5	Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio) - Primeira Simulação com o Algoritmo Clonal	66
4.6	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Primeira Simulação com o Algoritmo Clonal	66
4.7	Redes obtidas pela otimização via Algoritmo Clonal - Segunda Simulação.	67
4.8	Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio) - Segunda Simulação com o Algoritmo Clonal	67
4.9	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade) - Segunda Simulação com o Algoritmo Clonal	68
4.10	Redes obtidas pela otimização via AM1 - Primeira Simulação	70
4.11	Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Primeira Simulação com o AM1	70
4.12	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Primeira Simulação com o AM1	71
4.13	Redes obtidas pela otimização via AM1 - Segunda Simulação	71
4.14	Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Segunda Simulação com o AM1	72

4.15	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Segunda Simulação com o AM1	72
4.16	Redes obtidas pela otimização via AM2 - Primeira Simulação	75
4.17	Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Primeira Simulação com o AM2	75
4.18	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Primeira Simulação com o AM2	76
4.19	Redes obtidas pela otimização via AM2 - Segunda Simulação	76
4.20	Rede escolhida para ser aplicada ao projeto (priorizando custo para cenário médio)- Segunda Simulação com o AM2	77
4.21	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Segunda Simulação com o AM2	77
4.22	Redes obtidas pela otimização via AM3 - Primeira Simulação	80
4.23	Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Primeira e Segunda Simulação com o AM3	80
4.24	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Primeira Simulação com o AM3	81
4.25	Redes obtidas pela otimização via AM3 - Segunda Simulação	81
4.26	Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Segunda Simulação com o AM3	82
4.27	Soluções encontradas nas execuções realizadas pelos algoritmos de otimização	83
4.28	Comparação dos resultados obtidos pelos algoritmos de otimização	86
4.29	Sistema de 100 nós - Codificação por <i>Controlled-Greedy Encoding</i>	87
4.30	Redes obtidas pela otimização via Algoritmo Clonal para o caso de 100 nós	89
4.31	Rede ótima para o cenário mais provável para o caso de 100 nós - Algoritmo Clonal	89
4.32	Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando custo para o cenário médio)- Algoritmo Clonal	90
4.33	Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando confiabilidade) - Algoritmo Clonal	90
4.34	Redes obtidas pela otimização via AM3 para o caso de 100 nós	92
4.35	Rede ótima para o cenário mais provável para o caso de 100 nós - AM3	92
4.36	Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando custo para o cenário médio)- AM3	93
4.37	Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando confiabilidade) - AM3	93
4.38	Comparação dos resultados obtidos pelos algoritmos de otimização para o caso de 100 nós	95

Lista de Tabelas

1.1	Distância normalizada entre as redes exemplos	17
4.1	Parâmetros das distribuições de probabilidades	59
4.2	Dados dos condutores utilizados no projeto de redes de distribuição . . .	60
4.3	Parâmetros utilizados para a implementação do Algoritmo Clonal . . .	63
4.4	Soluções praticáveis - Simulação com o Algoritmo Clonal	63
4.5	Soluções praticáveis - Simulação com o AM1	69
4.6	Soluções praticáveis - Simulação com o AM2	74
4.7	Soluções praticáveis - Simulação com o AM3	79
4.8	Média da função objetivo para o cenário mais provável	83
4.9	Soluções não-dominadas resultantes do agrupamento entre respostas dos algoritmos	84
4.10	Dados das execuções realizadas	85
4.11	Soluções praticáveis - caso de 100 nós	88
A.1	Soluções praticáveis - Primeira Simulação com o Algoritmo Clonal . . .	107
A.2	Soluções praticáveis - Segunda Simulação com o Algoritmo Clonal . . .	108
A.3	Soluções praticáveis - Primeira Simulação com o AM1	109
A.4	Soluções praticáveis - Segunda Simulação com o AM1	110
A.5	Soluções praticáveis - Primeira Simulação com o AM2	111
A.6	Soluções praticáveis - Segunda Simulação com o AM2	112
A.7	Soluções praticáveis - Primeira Simulação com o AM3	113
A.8	Soluções praticáveis - Segunda Simulação com o AM3	114
A.9	Soluções praticáveis (caso de 100 nós)- Simulação com o Algoritmo Clonal	115
A.10	Soluções praticáveis (caso de 100 nós)- Simulação com o AM3	116

Lista de Acrônimos e Variáveis

A : conjunto de arestas de um grafo;

a : uma aresta do conjunto A ;

AC : Algoritmo Clonal;

AE : Algoritmo Evolucionário;

AG : Algoritmo Genético;

AM : Algoritmo Memético;

B : fração do total de anticorpos que é selecionada para a maturação de afinidade;

β : constante que define o número de clones gerados para cada anticorpo;

DL : distância local;

\vec{e}_i : é o i -ésimo vetor da base canônica do espaço \mathbb{R}^m ;

f^{mc} : função objetivo para o projeto de redes de distribuição de energia elétrica;

f_1 : custo da rede para o cenário mais provável;

f_2 : taxa de inactibilidade da rede;

f_3 : custo médio da rede para os cenários em que ela é factível;

f_4 : custo médio de falhas da rede nos cenários em que ela é factível;

G : grafo;

G_c : grafo completo;

M_A^G : matriz de adjacências do grafo G ;

m : número de arestas de um grafo;

MBL : Método de Busca Local;

mnv : número máximo de nós a que cada nó pode se conectar na *Controlled-Greedy Encoding*;

- mxv : número mínimo de nós a que cada nó pode se conectar na *Controlled-Greedy Encoding*;
- n : números de nós de um grafo;
- m : número de conexões possíveis em uma codificação;
- nc_i : número de conexões admissíveis para cada nó i ;
- n_{rp} : número de redes possíveis para um problema com t_c tipos de conexões;
- N_{pop} : número de anticorpos na população do Algoritmo Clonal;
- N : codificação para uma rede qualquer;
- \vec{N} : vetor representativo de uma rede N no espaço \mathbb{R}^m ;
- N_i : tipo de ramo que faz a conexão i da rede N ;
- \vec{N}_{AB} : vetor posição relativa entre duas redes N_A e N_B ;
- NC_i : número de clones gerados para o anticorpo i ;
- O : rede que representa a origem do espaço de redes;
- \vec{O} : representação de O no espaço \mathbb{R}^m ;
- pn_1 : fator de penalidade referente à topologia na métrica T -norm;
- pn_2 : fator de penalidade referente ao tipo de conexão na métrica T -norm;
- r_b : raio base de mutação;
- RNA : Redes Neurais Artificiais;
- SIA : Sistema Imunológico Artificial;
- SI : Sistema Imunológico;
- t_c : número de tipos de conexões possíveis para as conexões;
- V : conjunto de vértices de um grafo;
- w_i^N : peso da conexão i na rede N .

Capítulo 1

Projeto de Redes de Distribuição de Energia Elétrica

1.1 Introdução ao Projeto de Redes de Distribuição de Energia Elétrica

A utilização de técnicas de otimização para o projeto de redes de distribuição de energia é amplamente justificado, dada sua importância econômica e social. Cerca de 85% da população brasileira é consumidora de energia elétrica, que corresponde a 40% de toda a energia consumida no país (Soares, 2001). Há também o fato que o sistema de distribuição é responsável pela maior parte das perdas ocorridas no sistema elétrico e os recursos disponíveis para manutenção e expansão do mesmo são limitados.

Redes de distribuição de energia têm que ser constantemente expandidas ou re-projetadas, para que continuem atendendo às mudanças nas cargas, durante o decorrer dos anos (Carrano et al., 2005). O projeto dessas redes têm que levar em consideração não apenas a carga atual, mas também a carga que é esperada para um dado horizonte de tempo. Isto significa que este projeto tem incertezas intrínsecas já que geralmente não se consegue prever com exatidão os acréscimos de carga a serem inseridos no sistema para um horizonte a longo prazo. Adotar uma rede que foi projetada para uma carga sub-estimada significa que rapidamente o sistema não será capaz de atender à demanda

e um novo projeto deverá ser executado. Por outro lado, uma sobre-estimação da carga levará a uma rede de capacidade maior que a necessária e ela operará com parte de sua capacidade ociosa.

A variância da distribuição de probabilidade que representa as variáveis sujeitas à incertezas cresce ao longo do tempo. Como cada variáveis incerta pode assumir um conjunto de valores, um conjunto combinatorial de cenários de condições de operação para o sistema pode ser formado, sendo que cada cenário corresponde a um arranjo do estado em que cada variável do sistema pode apresentar. O conjunto de cenários cresce exponencialmente com o número de nós. Isto torna inviável a otimização da configuração da rede considerando todo o conjunto de cenários de carga possíveis. Encontrar a solução ótima para este problema é uma tarefa árdua, mesmo para sistemas com poucos nós. Por isso a maioria das metodologias considera um único possível cenário (o mais provável), ao invés de considerar um conjunto de possíveis cenários (Carrano et al., 2005). É utilizada neste trabalho, para a expansão da rede de distribuição, uma previsão da demanda para um dado horizonte de tempo. É considerado um cenário de carga médio ou mais provável, levando em conta apenas a média de crescimento anual da carga para prever o cenário futuro. A otimização do projeto de redes de distribuição de energia elétrica é realizada sobre o cenário mais provável.

O projeto de redes de distribuição é um problema complexo: combinatório e com funções não-lineares. Este projeto consiste em encontrar a configuração ótima da rede, que inclui a topologia (pares de nós que serão conectados) e o “setup” dos condutores (capacidade de cada condutor numa topologia específica), sujeito à restrições técnicas como atendimento à demanda e capacidade das linhas de transmissão. Este tipo de problema apresenta várias dificuldades de solução, mesmo em situações nas quais simplificações são consideradas (Pierre, 1993).

Dada sua natureza combinatória, existe um número muito grande de possíveis soluções para o problema e cada tipo de rede deve atender a características particulares. Esta característica das soluções no projeto de redes reduz o número de algoritmos que

podem ser utilizados. Os métodos determinísticos contínuos para otimização não-linear, que são baseados em direções de busca e exclusão de semi-espacos, não são recomendados neste caso. Eles dependem de cálculos da derivada, que não existem no espaço das soluções do problema de redes, que é discreto.

Técnicas que montam árvores de possibilidades garantem a obtenção do ótimo global, como o Branch-and-Bound (Vanderbei, 2001). No entanto, por terem custo computacional exponencial, essas técnicas tornam-se impraticáveis.

Algoritmos lineares para otimização de grafos, como Dijkstra (caminhos mínimos), Kruskal (árvore geradora mínima) e Ford-Fulkerson (fluxo máximo) apresentam grande eficiência para os problemas a que foram propostos, tendo aplicação restringida para outros problemas (Dijkstra, 1959; Ahuja et al., 1993; Bazaraa et al., 1991). Versões discretas do Simplex (Vanderbei, 2001) podem, também, ser aplicadas à problemas de redes, mas para isto as funções devem ser lineares. Isto faz com que problemas não-lineares tenham que ser aproximados por funções de primeira ordem, o que geralmente compromete o resultado final.

A restrição ao uso da maior parte das técnicas conforme foi descrito, implica no estudo de alternativas para a solução do problema de redes. Isto motiva a utilização de métodos heurísticos na solução de problemas não-lineares. Os algoritmos evolucionários, que são algoritmos heurísticos, aparecem como ferramentas de destaque, principalmente por sua flexibilidade e robustez. A flexibilidade e robustez se dão principalmente pela forma com que os algoritmos evolucionários são construídos, sem premissas matemáticas fortes como linearidade, diferenciabilidade ou convexidade. Dadas estas características, existem várias aplicações destes algoritmos nos mais variados problemas relacionados à redes.

Algoritmos baseados em sistemas imunológicos artificiais são uma proposta recente de algoritmos evolucionários, do fim da década de 90 (Dasgupta, 1998; de Castro and Timis, 2002; Dasgupta et al., 2003). Os Sistemas Imunológicos Artificiais (SIA) não respondem somente com uma única solução, mas um conjunto de soluções, formado

pela solução ótima e pelas soluções sub-ótimas que são mantidas e evoluídas ao longo da execução do algoritmo (de Castro and Von Zuben, 2002; de Castro, 2001), que é uma característica muito importante destes algoritmos. Por isso o algoritmo de seleção clonal ou simplesmente Algoritmo Clonal (AC), que é uma ferramenta dos SIA, é proposto aqui para tratar o problema de otimização de redes de distribuição de energia. A metodologia utilizada neste trabalho para tornar as redes projetadas robustas na presença das incertezas, necessita não somente da solução ótima do projeto de redes como também das soluções sub-ótimas .

A metodologia proposta consiste em avaliar as soluções obtidas pelo algoritmo de otimização em um conjunto de cenários distintos do cenário mais provável, considerado no projeto. Este conjunto de cenários é obtido pelo Método de Monte Carlo, onde as incertezas consideradas, que são a evolução no crescimento da carga e as variações no preço da energia elétrica ao longo do tempo, são modeladas por distribuições de probabilidade para cada nó do sistema. Para cada cenário obtido são avaliadas todas as soluções ótimas quanto ao seu desempenho para os critérios estabelecidos.

Uma importante limitação dos algoritmos evolucionários, entre eles o algoritmo clonal, é que embora sejam ferramentas capazes de encontrar uma aproximação do ótimo global do problema, eles pecam no sentido da precisão da busca pelo ótimo (Wanner, 2006). Uma maneira de melhorar o desempenho destes algoritmos é a incorporação de métodos capazes de realizar uma busca local sobre as soluções obtidas pelo algoritmo. Os algoritmos provenientes da hibridação entre um algoritmo e os algoritmos evolucionários são geralmente chamados de algoritmos meméticos (AM), embora outras denominações também sejam utilizadas. Para a implementação dos algoritmos meméticos utilizados, foram utilizados o algoritmo de seleção clonal e métodos de busca local desenvolvidos. Foram propostos neste trabalho dois métodos de busca local para o espaço de redes implementados à partir da métrica *T-norm* proposta por Carrano et al. (2007b). Estes métodos foram incorporados ao algoritmo clonal em estágios distintos da execução do mesmo e em combinações de mais de um método de busca local, dando

horigem à três algoritmos meméticos para o projeto de redes de distribuição de energia.

Os resultados obtidos mostram que a utilização de algoritmos meméticos, quando comparados com os resultados obtidos pelo algoritmo clonal, são em geral melhores. O uso dos algoritmos meméticos possibilita a escolha de redes que apresentam um custo satisfatório comparado com o custo da rede do cenário mais provável. Além disso as redes com o uso dos meméticos apresentam baixas taxas de infactibilidade, baixos custos para os cenários para os quais elas são factíveis e baixos custos de falhas. Elas são portanto redes mais robustas e mais confiáveis.

1.2 Representação das Variáveis do Projeto de Redes de Distribuição de Energia Elétrica

1.2.1 Grafos

Um grafo G consiste de um conjunto finito V de vértices (ou nós), um conjunto finito A de arestas (ou conexões ou ramos), e uma matriz de adjacência M_A , que associa a cada aresta a de G um par não ordenado de vértices de G (não necessariamente distintos), chamados de extremos de a (Bondy and Murty, 1976; Wilson, 1996).

Dependendo da aplicação, podem-se ter arestas direcionadas e que ligam um vértice a ele próprio (realimentação), além de ser possível a associação de pesos numéricos aos vértices e arestas. Os grafos utilizados para representar as redes de distribuição de energia elétrica são não-direcionais e sem pesos associados às conexões, representados por grafos planares estruturados como árvores.

A figura 1.1 apresenta um exemplo de grafo não-direcional, $G(V, A)$, com seis vértices ($V = [1, 2, 3, 4, 5, 6]$) e 10 arestas ($A = [a, b, c, d, e, f, g, h, i, j]$)¹.

O estabelecimento de caminhos fechados no grafo (*loops*) não é permitido. Com isso, o conjunto de arestas possíveis (grafo completo ou G_C) no projeto de uma rede de

¹As figuras 1.1, 1.2, 1.3 e 1.5 foram obtidas de Carrano (2007)

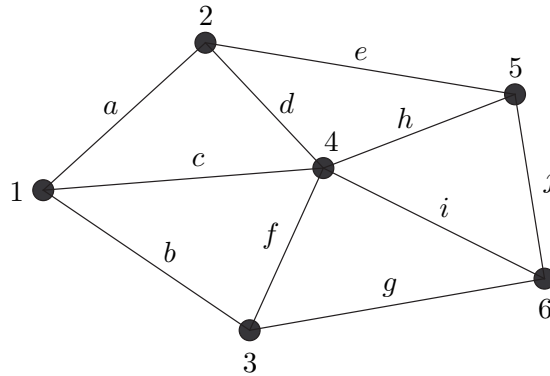


Figura 1.1: Exemplo de grafo

n nós pode ser representado pela matriz de adjacência M_A apresentada na Eq. (1.1). A Eq. (1.2) determina o número de arestas (m) em G_C . A figura 1.2 apresenta o grafo completo para o conjunto de vértices do grafo da figura 1.1, que é um sub-grafo de G_C .

$$M_A = \begin{cases} a_{ii} = 0 & i = 1, \dots, n \\ a_{ij} = 1 \forall i \neq j \end{cases} \quad (1.1)$$

$$m = \frac{n \cdot (n - 1)}{2} \quad (1.2)$$

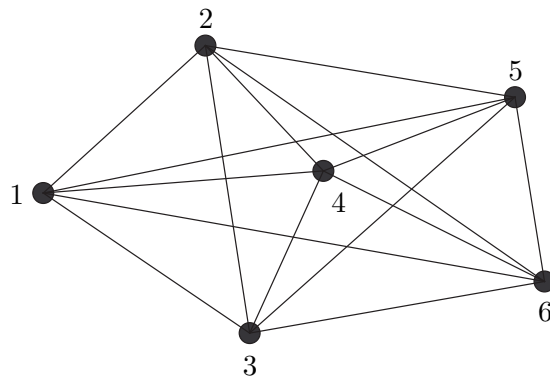


Figura 1.2: Grafo completo para o exemplo da figura 1.1

Árvores

As árvores representam umas das classes mais importantes de grafos, uma vez que são aplicadas na solução de grande parte dos problemas práticos.

Sendo $G(V, A)$ um grafo qualquer, pode-se definir que G é uma árvore, se e somente se, G é um grafo conexo que não contém *loops*. Dessa definição, derivam alguns teoremas importantes sobre árvores (Narsingh, 1984):

Teorema 1.1 *Existe um e apenas um caminho, entre qualquer par de vértices em uma árvore.*

Teorema 1.2 *Uma árvore com n vértices tem $n - 1$ arestas.*

As demonstrações desses teoremas podem ser encontradas na referência (Narsingh, 1984). Uma característica interessante das árvores é que a remoção ou adição de um ramo na mesma faz com que ela deixe de ser uma árvore. A remoção de uma aresta faz com que o grafo deixe de ser conexo e a adição de uma aresta insere um *loop* no grafo. Outro importante teorema sobre grafos em árvore é apresentado em Cayley (1989):

Teorema 1.3 Teorema de Cayley *Em um grafo completo G_c , com n vértices e $m = \frac{n \cdot (n - 1)}{2}$ arestas, existem n^{n-2} árvores que são sub-grafos de G_c .*

O teorema de Cayley nos permite ter uma dimensão do número de soluções possíveis para o problema de redes representadas por grafos em árvore. Com o aumento do número de nós, o total de redes que podem ser obtidas aumenta exponencialmente.

A figura 1.3 ilustra um exemplo de árvore.

1.2.2 Representação das Variáveis

No projeto de redes, o grafo completo G_C define o espaço de busca do problema de otimização. Isso significa que cada aresta do grafo (que também é chamada de conexão ou ramo neste trabalho) representa uma variável do problema. Em termos práticos, o

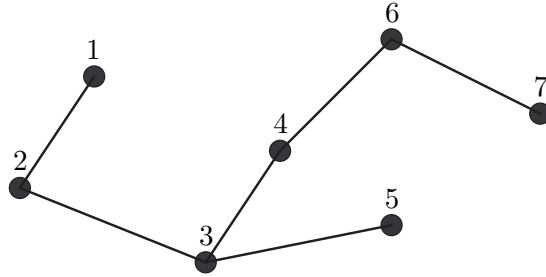


Figura 1.3: Exemplo de árvore

projeto de redes consiste em buscar um sub-grafo de G_C que atenda determinadas características estruturais (devem ser árvores e atender às restrições técnicas do problema tratado) e seja satisfatório dentro de um ou mais critérios estabelecidos para projeto.

A Eq. (1.3) apresenta a lista de conexões possíveis (ou codificação) em um problema de n nós e o vetor X que representa a codificação das redes neste problema. Neste caso, se $x_i = 1$, significa que os nós referentes à conexão i estão conectados e se $x_i = 0$, não existe conexão entre eles.

$$\begin{array}{l}
 \text{de} \quad 1 \quad 1 \quad \dots \quad 2 \quad 2 \quad \dots \quad n-1 \\
 \text{para} \quad 2 \quad 3 \quad \dots \quad 3 \quad 4 \quad \dots \quad n \quad x_i \in \mathbb{Z} / x_i \in [0, 1] \quad (1.3) \\
 X = [x_1 \quad x_2 \quad \dots \quad x_n \quad x_{n+1} \quad \dots \quad x_m]
 \end{array}$$

Onde:

m é o número de conexões possíveis.

Entretanto, a codificação binária na forma em que é apresentada na Eq. (1.3) não é suficiente para o projeto de redes de distribuição tratado neste trabalho. Aqui uma determinada conexão pode assumir alguns valores distintos, ou seja, cada conexão pode ser realizada por um determinado tipo de cabo e cada cabo tem características técnicas próprias. Uma alternativa viável para solucionar este problema é a utilização da representação proposta em (Ramírez-Rosado and Bernal-Agustín, 1998). Nesta codifi-

cação, cada conexão possível pode assumir valores inteiros que variam de 0 (ausência de conexão) à t_c (nós conectados com uma conexão de tipo t_c , onde t_c é o número de tipos de conexões possíveis). A Eq. (1.4) ilustra essa codificação:

$$\begin{array}{l}
 \text{de} \quad 1 \quad 1 \quad \dots \quad 2 \quad 2 \quad \dots \quad n_n - 1 \\
 \text{para} \quad 2 \quad 3 \quad \dots \quad 3 \quad 4 \quad \dots \quad n_n \quad x_i \in \mathbb{Z} / x_i \in [0, t_c] \quad (1.4) \\
 X = [\quad x_1 \quad x_2 \quad \dots \quad x_n \quad x_{n+1} \quad \dots \quad x_m \quad]
 \end{array}$$

1.3 Controle Dimensional em Problemas de Redes

O número de conexões possíveis em um problema de redes (ou codificação) cresce de forma quadrática com o número de nós. Esse comportamento faz com que a solução direta de problemas de média e grande dimensão apresente custo computacional elevado, devido ao alto número de variáveis a serem consideradas.

Conhecimentos *a priori* do problema tratado podem ser aplicados para reduzir o conjunto de conexões possíveis. Em problemas de distribuição de energia elétrica por exemplo, devem-se ignorar conexões entre nós que se encontrem separados por acidentes geográficos como lagos, montanhas etc.. A instalação de linhas de distribuição nesses locais é inviável tanto econômica quanto tecnicamente.

No entanto a exclusão dessas conexões geralmente não é suficiente para a solução de problemas de grande porte. Nas abordagens encontradas na bibliografia, como Miranda et al. (1994); Ramírez-Rosado and Bernal-Agustín (1998); Cossi et al. (2005), o conjunto de conexões viáveis para o problema é estabelecido manualmente, baseado na experiência do projetista. Este procedimento pode ser executado de duas maneiras:

Inclusão de Conexões: partindo de um conjunto de conexões vazio, inclui-se uma a uma as conexões consideradas viáveis para o problema; e

Exclusão de Conexões: partindo de um conjunto de conexões com todas as conexões possíveis, exclui-se uma a uma as conexões consideradas inviáveis para o problema.

Apesar de o método de Inclusão de Conexões ser menos trabalhoso para o projetista, o mesmo pode levar a soluções inadequadas, uma vez que a não inclusão de uma conexão presente na solução ótima exclui a possibilidade de se encontrar a mesma. Já o método de Exclusão de Conexões permite a obtenção da solução ótima, sendo porém muito mais trabalhoso.

Para contornar a dificuldade de se fazer a codificação do problema de redes de distribuição de energia elétrica, tornando este processo automático, foi utilizada a técnica *Controlled-Greedy Encoding* de redução dimensional da codificação do problema (Carrano, 2007; Carrano et al., 2006). Essa técnica não depende de nenhum conhecimento *a priori*, mas não exclui a utilização desses conhecimentos em uma etapa subsequente à execução da mesma. O resultado da aplicação dessa técnica é um número bem menor de variáveis do que aquele obtido pela codificação completa. É ainda possível a aplicação do método de Exclusão de Conexões (para tornar a dimensão do problema ainda menor) ou a Inclusão de Conexões (para acrescentar conexões que por ventura não tenham sido incluídos pela *Controlled-Greedy Encoding*) e que são fundamentais para o projeto.

1.3.1 *Controlled-Greedy Encoding*

A construção do *Controlled-Greedy Encoding* (Carrano et al., 2006) foi inspirada em algoritmos gulosos (*greedy algorithms*) cuja idéia básica é conectar cada nó ao nó mais próximo, o que geralmente conduz à ótimos locais. Para evitar isso buscou-se controlar a “gula” do algoritmo utilizando 2 parâmetros:

- *mnv*: número mínimo de nós a que cada nó pode se conectar;
- *mxv*: número máximo de nós a que cada nó pode se conectar.

Para determinar o número de conexões de cada nó determina-se primeiramente a distância média de cada nó i (d_i), utilizando (1.5).

$$d_i = \frac{1}{n} \sum_{j=1}^n \text{dist}(i, j) \quad (1.5)$$

O nó com menor distância média deve se conectar com mxv nós e o nó com maior distância média deve se conectar com mnv nós. Para os nós intermediários foi proposta a utilização de uma função linear discretizada, conforme mostra a figura 1.4. Com isso, o número de conexões admissíveis para cada nó i (nc_i) pode ser determinado usando (1.6).

$$nc_i = \left\lfloor \left(\frac{mnv - mxv}{d_{max} - d_{min}} \right) \cdot (d_i + d_{min}) + mxv \right\rfloor \quad (1.6)$$

onde:

$$d_{min} = \min(d)$$

$$d_{max} = \max(d)$$

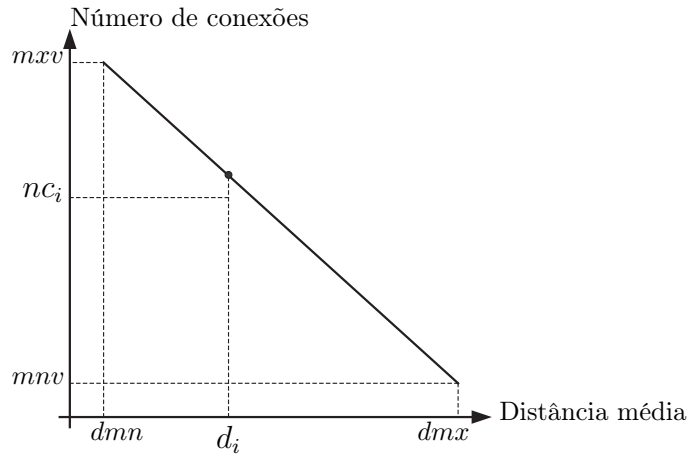


Figura 1.4: Número de conexões *vs.* distância média

A seguir é apresentado o procedimento desta codificação descrita:

Controlled-Greedy Encoding

for i **from** 1 **to** n **do**

$cn_i \leftarrow \emptyset$;

$$d_i \leftarrow \frac{1}{n} \cdot \sum_{j=1}^{n_n} dist(i, j);$$

$$nc_i \leftarrow \left\lfloor \left(\frac{mnv - mxv}{d_{max} - d_{min}} \right) \cdot d_i + mxv + d_{min} \cdot \left(\frac{mxv - mnv}{d_{max} - d_{min}} \right) \right\rfloor;$$

for j **from** 1 **to** nc_i **do**

$$cn_i \leftarrow cn_i \cup vz_j;$$

end for

end for

1.4 Representação das Redes no Espaço Contínuo

O projeto de redes pode ser dividido em duas partes, que são:

1. A busca da topologia ótima da rede (definição de quais são os nós que se conectam entre si);
2. A busca do tipo ótimo de ramo para cada conexão (definição de qual o tipo de cabo a ser utilizado em cada conexão).

Como existe uma forte ligação entre estas duas partes do problema, elas não podem ser dissociadas para serem tratadas independentemente. O desempenho da rede é medido por uma função não-linear da sua estrutura e dos tipos de conexões. Vários são os exemplos de aplicações dos problemas que envolvem redes: redes de energia elétrica, redes de transportes e outras. Nestes problemas, qualquer mudança na topologia ou no tipo de ramo afeta completamente todo o fluxo na rede. Mesmo o fluxo total pode mudar em alguns casos, sem obedecer alguma restrição de conservação de fluxo - este é o caso, por exemplo, do fluxo de energia na rede de distribuição. Nestes casos, a função custo se torna uma função não-linear da topologia da rede e do fluxo resultante (Carrano, 2007).

Portanto, o projeto de redes deve ser tratado considerando os dois critérios principais ao mesmo tempo, uma vez que ambos interferem indissoviavelmente no desempenho da rede.

Os algoritmos evolucionários são uma boa ferramenta para lidar com este tipo de problema. Os seus operadores podem ser adaptados para trabalhar no sentido de ao mesmo tempo buscar pela estrutura ótima da rede e pelo tipo ótimo de conexão. As perturbações feitas sobre as soluções para o caso contínuo sempre levam à soluções que têm sentido real, ou seja, permanecem fazendo sentido físico para o problema, mesmo que com propriedades muito diferentes da original. Quando as mesmas perturbações são realizadas sobre uma solução de um problema discreto, elas podem levar à redes que não fazem nenhum sentido prático. Por exemplo, na codificação apresentada em (1.4), uma alteração num dado elemento x_i em uma rede factível (uma árvore), pode dar origem a uma rede desconexa ou acrescentar um *loop*. Esta rede é infactível, já que perde sua estrutura em árvore, não fazendo mais sentido físico para o projeto de redes (Smith and Walters, 2000; Carrano et al., 2006). A utilização de penalizações e tratamento das soluções infactíveis, para o caso não é uma alternativa muito atraente, pois pode levar aos seguintes resultados insatisfatórios:

- a propagação de soluções infactíveis, o que força a busca apenas sobre as poucas soluções factíveis, reduzindo a eficiência do algoritmo;
- a substituição das soluções infactíveis por novas soluções, implica em perda de eficiência do algoritmo e aumento do custo computacional, uma vez que as soluções novas não passaram por nenhum processo de melhoria;
- a substituição de soluções infactíveis por soluções factíveis já presentes na população, diminui a diversidade de busca, comprometendo o desempenho do algoritmo;
- o adequamento de soluções infactíveis para torná-las factíveis, pode ter um custo computacional muito elevado.

Dado que os métodos de penalização e tratamento de soluções infactíveis não são uma boa alternativa para tratar as soluções infactíveis no problema de redes de distribuição, uma alternativa é a implementação de operadores capazes de garantir que

as soluções obtidas sejam factíveis. Isto pode aumentar consideravelmente o desempenho dos algoritmos utilizados em problemas de redes (Carrano et al., 2006). Para a construção de operadores capazes de gerar redes factíveis, com operações baseadas em propriedades do espaço contínuo, foi proposto em (Carrano et al., 2007b), uma representação vetorial para as redes. Esta representação, através dos conceitos de espaços métricos (Lima, 1995), permite uma representação das redes no espaço contínuo permitindo assim a utilização de propriedades deste espaço, sendo desta maneira cada rede representada por pontos discretos no espaço \mathbb{R}^m , onde m é a dimensão deste espaço (número de variáveis). Esta representação “contínua” das redes é baseada em uma definição adequada de norma, chamada *T-norm*, que é definida a partir de um produto escalar, conforme será visto adiante. Com a representação no espaço contínuo é possível definir conceitos como posição relativa, distância, direções, vizinhança, entre outros, para as redes.

1.4.1 O Conceito de Distância no Espaço de Redes - *Métrica T-norm*

Os conceitos apresentados a seguir foram adaptados de Carrano (2007), onde uma definição matemática mais completa a respeito da métrica *T-norm* é apresentada.

Seja N uma rede qualquer que pode ser representada em um grafo $G(V, A)$, sendo $|V| = n$ e $|A| = m$, com t_c tipos de conexões. Dados os conceitos de espaços métricos (Lima, 1995), pode-se representar N como um vetor \vec{N} no espaço \mathbb{R}^m como em (1.7):

$$\vec{N} = \sum_{i=1}^m (w_i^N \cdot pn_1 + N_i \cdot w_i^N \cdot pn_2) \cdot \vec{e}_i \quad (1.7)$$

onde:

w_i^N é o peso da conexão i na rede N ;

N_i é o tipo de ramo da conexão i na rede N ;

pn_1 é o fator de penalidade 1;

pn_2 é o fator de penalidade 2;

\vec{e}_i é o i -ésimo vetor da base canônica do espaço \mathbb{R}^m .

A equação (1.7) representa uma rede qualquer N no espaço \mathbb{R}^m , sendo esta rede um ponto neste espaço. Deste modo, todas as propriedades do espaço contínuo podem ser empregadas sobre a representação da rede N . Sejam as redes N_A e N_B redes quaisquer, tais que suas representações no espaço contínuo sejam:

$$\vec{N}_A = \sum_{i=1}^m (w_i^A \cdot pn_1 + A_i \cdot w_i^A \cdot pn_2) \cdot \vec{e}_i \quad (1.8)$$

$$\vec{N}_B = \sum_{i=1}^m (w_i^B \cdot pn_1 + B_i \cdot w_i^B \cdot pn_2) \cdot \vec{e}_i \quad (1.9)$$

O vetor posição relativa entre N_A e N_B é uma consequência direta desta representação vetorial das redes, conform apresentado em 1.10:

$$\vec{N}_{AB} = \sum_{i=1}^m [pn_1 \cdot (w_i^A - w_i^B) + pn_2 \cdot (A_i \cdot w_i^A - B_i \cdot w_i^B)] \cdot \vec{e}_i \quad (1.10)$$

Um escalar que define a distância entre duas redes pode ser calculado através de qualquer norma. A medida de distância que utiliza a norma euclidiana para calcular a distância da equação (1.10) será chamada ao longo deste texto por métrica T -norm. A expressão para se calcular a T -norm é apresentada na equação (1.11):

$$dist(N_A, N_B) = \left\{ \sum_{i=1}^m [pn_1 \cdot (w_i^A - w_i^B) + pn_2 \cdot (A_i \cdot w_i^A - B_i \cdot w_i^B)]^2 \right\}^{\frac{1}{2}} \quad (1.11)$$

As demonstrações necessárias para verificar que a representação de redes no espaço contínuo e que a métrica T -norm são válidas podem ser encontradas em (Carrano, 2007).

Pode-se ver claramente que (1.10) considera os dois sub-problemas citados anteriormente: o termo $pn_1 \cdot (w_i^A - w_i^B)$ calcula a parcela do vetor referente a diferenças na topologia e o termo $pn_2 \cdot (A_i \cdot w_i^A - B_i \cdot w_i^B)$ calcula a parcela do vetor relacionada com a diferenciação de ramos.

A forma de calcular os pesos das conexões (w_i^N) deve ser analisada para cada estrutura de rede. Em redes em árvore, que serão tratadas ao longo deste trabalho, o peso da conexão i (que conecta os nós a e b do espaço onde ambas as redes são definidas) pode ser calculado através da Eq. (1.12):

$$w_i^N = \frac{s_a^N - s_b^N}{2} \quad (1.12)$$

$$s_x^N = 1 - \frac{d_{x,raiz}^N}{\max(d_{j,raiz}^N)} \quad (1.13)$$

onde:

a e b são os nós extremos da conexão i na rede N ;

s_x^N é o peso do nó x na rede N ;

$d_{x,raiz}^N$ é a distância total do caminho entre o nó x e o nó raiz da rede N (medida em km);

$\max(d_{j,raiz}^N)$ é a máxima distância dos caminhos entre um nó j e a raiz da rede N , para todos os j pertencentes ao conjunto V .

O peso w_i representa o fato de que conexões próximas à raiz da rede têm influência maior na caracterização da rede do que conexões próximas às extremidades da rede. Isto significa que uma mudança em uma conexão próxima à raiz causa uma grande modificação no vetor que representa a rede, enquanto uma mudança em uma conexão extrema da rede causaria uma modificação pequena no vetor que representa a mesma. Isto é o que se espera de uma rede de distribuição de energia: o fluxo de potência sofre uma grande perturbação se ocorrer uma mudança nas conexões próximas à raiz da rede enquanto que uma mudança nas conexões terminais da rede provoca pouca alteração no fluxo de potência. Os coeficientes pn_1 e pn_2 devem ser ajustados de modo que redes

com diferentes topologias sejam “mais diferentes” do que redes com a mesma topologia e apenas com ramos diferentes. Assim, pn_1 pondera a diferença de topologia entre as redes e pn_2 pondera a diferença entre os tipos de ramos. Para o caso deste trabalho sugere-se $pn_1 = k \cdot t_c$ e $pn_2 = 1$, onde k é uma constante e t_c é o número de tipos de ramo. Neste caso, uma mudança na topologia tem um impacto k vezes maior que a mudança no tipo de ramo.

Quatro redes foram geradas aleatoriamente para ilustrar a métrica T -norm (figura 1.5). A Tab. 1.1 apresenta a distância normalizada entre essas redes. As redes A e B são bem similares, tendo a mesma topologia e diferindo apenas nos tipos de ramos. Na tabela pode-se ver que a distância entre as mesmas é pequena. Redes que apresentam várias diferenças na topologia (como A e D ou B e C) têm grande distância entre si. A distância entre as redes A e C é próxima à distância entre as redes B e C , o que é esperado, dado a similaridade entre A e B . Com base nessas observações percebe-se que a métrica proposta em (1.11) tem o comportamento esperado de uma medida de distância, o que dá suporte à sua aplicação (Carrano, 2007).

Tabela 1.1: Distância normalizada entre as redes exemplos

Rede	A	B	C	D
A	0	0.362	0.610	1.000
B	0.362	0	0.614	0.994
C	0.610	0.614	0	0.931
D	1.000	0.994	0.931	0

1.4.2 Geração de Redes Aleatórias à Distâncias Pré-definidas

A geração aleatória de uma rede que se encontra a uma distância pré-definida de uma dada rede é uma operação fundamental para os algoritmos utilizados neste trabalho. Os operadores de hipermutação e de busca local, utilizados nos algoritmos, foram desenvolvidos com base neste princípio.

Seja uma rede inicial P e uma dada distância requerida γ . Para definir uma rede R que esteja à distância γ de P , deve-se seguir os seguintes passos:

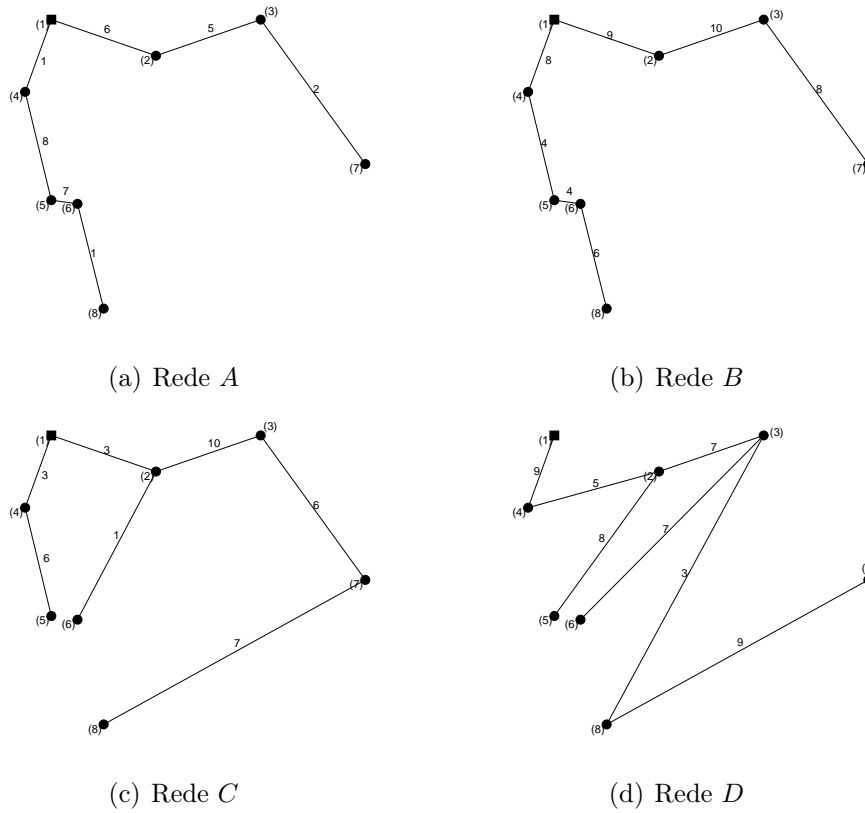


Figura 1.5: Redes exemplos para a métrica T -norm

Geração de redes aleatórias à distâncias pré-definidas

```

conf ← 1;
 $\vec{R}_x \leftarrow \vec{P}$ ;
if conf = 1 then
     $C_{\vec{R}_x} \leftarrow \text{conj. removidas}(\vec{R}_x)$ ;
    if  $C_{\vec{R}_x}$  não é vazio then
         $A \leftarrow \text{aleatorio}(C_{\vec{R}_x})$ ;
         $\vec{R}_x \leftarrow \text{modificação1}(\vec{R}_x, A)$ ;
        conf = 1;
    else
        conf = 0;
    end if
end if

```

```

 $CC_{\vec{R}_x} \leftarrow \mathbf{conj. tipos}(\vec{R}_x);$ 
 $\vec{R} \leftarrow \vec{R}_x;$ 
for  $i = 1$  até  $\|CC_{\vec{R}_x}\|$  do
     $\vec{R}\vec{R} \leftarrow \mathbf{modificação2}(\vec{R}_x, CC_{\vec{R}_x}(i));$ 
    if  $T - norm(\vec{P}, \vec{R}\vec{R}) - \gamma \leq \epsilon$  then
         $\vec{R} \leftarrow \vec{R}\vec{R};$ 
        saia do for;
    end if
end for

```

onde:

\vec{P} : rede a partir da qual será gerada uma nova rede à distância γ pela métrica T -norm com uma tolerância ϵ ;

$C_{\vec{R}_x}$: conjunto de conexões de \vec{R}_x que podem ser removidas;

conj. removidas: determina o conjunto de conexões de \vec{R}_x que podem ser removidas, tal que a rede resultante R (após ser corrigida para manter-se factível) cumpra a desigualdade: $\|R, \vec{P}\| \leq \gamma$

A : variável auxiliar que define uma conexão escolhida aleatoriamente dentro do conjunto $C_{\vec{R}_x}$;

modificação1: remove a conexão definida em A de \vec{R}_x , mantendo a rede resultante factível, e copia o resultado para \vec{R}_X ;

conj. tipos: determina o conjunto de conexões de \vec{R}_x que podem ter seus tipos mudados, de forma que a rede resultante R cumpra $\|R, \vec{P}\| - \gamma \leq \epsilon$;

modificação2: muda o tipo de conexão da conexão i do conjunto $CC_{\vec{R}_x}$ da rede \vec{R}_x ;

$\vec{R}\vec{R}$: variável auxiliar;

\vec{R} : rede gerada à uma distância γ pela métrica T -norm com uma tolerância ϵ da rede \vec{P} .

1.5 Formulação do Problema de Redes de Distribuição de Energia Elétrica

Considerando a representação de redes apresentada na Sec. 1.2.2 e sendo $f_c(X)$ uma função de X que se deseja minimizar, pode-se formular o problema do projeto de redes da seguinte forma:

$$X^* = \underset{X}{\operatorname{argmin}} f_c(X) \quad (1.14)$$

$$\text{sujeito a } : X \in F_X \subset G_c \quad (1.15)$$

onde:

F_X é o conjunto de redes factíveis.

Três aspectos básicos devem ser considerados no projeto de redes de distribuição de energia elétrica (Willis et al., 1996):

- minimização do investimento para a instalação e/ou redimensionamento do sistema;
- minimização dos custos com manutenção da rede;
- minimização das perdas de energia elétrica na rede.

Além de objetivar a minimização dos custos citados a rede deve atender a requisitos técnicos de projeto, para que ela possa operar em conformidade com a qualidade e legislação quanto ao fornecimento da energia. As quatro restrições que devem ser obedecidas são:

- atendimento de todos os consumidores (todos os nós devem ter sua demanda de energia atendida);

- manutenção da estrutura da rede (é necessário que as redes tenham estrutura radial, o que significa manter a rede como uma representação em árvore);
- trabalhar sob o limite de capacidade de transferência de potência das linhas de transmissão da rede ;
- atender aos níveis de tensão regulamentados para cada barra de carga.

As duas primeiras restrições se referem à garantia de manutenção da estrutura da rede (em árvore com todos os nós conexos). As duas últimas se referem a aspectos técnicos particulares da rede de distribuição de energia elétrica.

Os três objetivos propostos para o problema podem ser agregados em uma única função objetivo, que corresponde ao custo total da rede. Eles podem ser agregados porque representam grandezas econômicas, diferindo-se apenas nos instantes de tempo em que os recursos são aplicados, conforme discutido em (Ramírez-Rosado and Bernal-Agustín, 1998; Carrano et al., 2006). Desta forma, as equações (1.16) e (1.17) representam a função objetivo para o projeto de redes de distribuição de energia elétrica e as restrições do problema.

$$f^{mc} = \sum_{i=1}^m Y_i^N \cdot IC(N_i) + \sum_{t=1}^{at} \left\{ \sum_{i=1}^m Y_i^N [MC(N_i) + LC_i] \right\} (1 - int^{rt})^{t-1} \quad (1.16)$$

$$\begin{aligned} c_1 : \sum_{i=1}^m Y_i^N &= |V| - 1 \\ c_2 : \sum_{i=1}^m Y_i^N &\leq |S| - 1 \\ c_3 : I_i &\leq I_{max}(N_i) \quad \forall i \in N \\ c_4 : 0,92 &\leq V_i^n \leq 1,08 \quad \forall i \in V \end{aligned} \quad (1.17)$$

onde:

N é a rede avaliada;

m é o número de conexões possíveis;

at é o tempo previsto para o projeto;

$f^{mc}(N)$ é o custo monetário no tempo presente da rede N (em \$);

Y_i^N é 1 se a conexão está presente na rede N ou 0 caso contrário;

N_i é o tipo de ramo utilizado na conexão i ;

$IC(N_i) = l_i.brc(N_i)$ é o custo total da instalação (ou substituição) do ramo i (em \$);

$MC(N_i) = l_i.mnc(N_i)$ é o custo total de manutenção do ramo i (em \$/ano);

$LC_i = 8760.l^f.en^{tax}.P_i^L$ é o custo total de perdas do ramo i (em \$/ano);

int^{rt} é a taxa de juros anual;

V é o conjunto de nós da rede N ;

S é o conjunto de nós induzido pelo conjunto de conexões N ;

I_i é a corrente no ramo i ;

V_i^n é a tensão no nó i ;

l_i é o comprimento da conexão i (em km);

$brc(N_i)$ é o custo do ramo de tipo N_i (em \$/ km);

$mnc(N_i)$ é o custo de manutenção do ramo de tipo N_i (em \$/ km /ano);

l^f é o fator de perda;

en^{tax} é a tarifa de energia (em \$/ kWh);

P_i^L é a perda de energia no ramo i (em kW).

Da equação (1.16) observa-se que a função objetivo é composta por duas parcelas principais. A primeira parcela são os custos fixos, compostos pelos custos com instalação, que são gastos imediatamente quando a rede é instalada. A segunda parcela são os custos variáveis, que são os custos com manutenção e perdas de energia. Estes custos ocorrem ao longo do tempo considerado para o projeto e podem ser trazidos para o momento presente para determinar o custo atual da rede. Os custos de instalação e manutenção são estritamente dependentes do comprimento e do tipo de cabo utilizado nas conexões. Já os custos com perdas de energia são relacionados à topologia da rede e à condição de operação considerada no projeto.

Desta maneira, este problema de otimização mono-objetivo pode ser descrito pela formulação dada na equação (1.14). Por meio da equação (1.16) pode-se notar a forte relação existente entre as variáveis do problema. Alterações nas conexões ou tipos de ramos afetam não-linearmente todo o fluxo de potência da rede, alterando a maior parte das perdas e correntes que fluem nas conexões. Por isto, tornam-se ineficientes as abordagens que tratam o problema por partes isoladas, obtendo primeiro a topologia para depois ajustar os tipos de conexão.

Capítulo 2

Algoritmos de Otimização

2.1 Descrição do Algoritmo de Seleção Clonal

2.1.1 Inspiração Biológica

O Sistema Imunológico

Vários aspectos da natureza já foram explorados por pesquisadores para servirem de inspiração para o desenvolvimento de algoritmos eficientes capazes de tratar os mais diversos problemas científicos. A teoria da evolução natural é utilizada como inspiração para a formulação de algoritmos evolucionários, com destaque para os Algoritmos Genéticos (Holland, 1975; Goldberg, 1989). Ela também inspirou os Sistemas Imunológicos Artificiais (SIA), que são baseados em teorias sobre os sistemas imunológicos dos seres vivos. Existe uma vasta literatura a respeito do sistema imunológico e através de adaptações dessas teorias é possível estruturar algoritmos eficientes para lidar com diversos problemas.

Os SIA têm atraído a atenção de diferentes comunidades, com os interesses mais distintos tais como reconhecimento de padrões, auto identificação, análise de dados, aprendizado de máquinas, detecção de anomalias, otimização, diagnóstico de faltas, reconhecimento de padrões, etc.. Os métodos baseados em sistemas imunológicos artificiais têm se mostrado eficientes na solução de problemas de otimização (X. Wang and

Ovaska, 2004).

Sistema Imunológico - Descrição

O sistema imunológico (SI) é um complexo de células, moléculas e organismos que representam um mecanismo de identificação capaz de perceber e combater disfunções em suas próprias células e ações de microorganismos infecciosos exteriores. Suas células e moléculas mantêm a sobrevivência dos organismos infectados. Eles reconhecem uma variedade muito grande de células e substâncias infecciosas exteriores, chamadas de elementos externos (*nonsel*), distinguindo-os das células nativas não-infecciosas, conhecidas com moléculas próprias (*self*) (Jerne, 1973; de Castro, 2001). Quando o patógeno (elemento infeccioso externo) entra no corpo, ele é detectado e levado à eliminação. O sistema é capaz de relembrar cada infecção, tal que uma nova infecção da mesma patologia é eliminada com maior eficiência.

Existem dois sistemas inter-relacionados pelos quais o corpo identifica materiais exteriores: o sistema imunológico inato e o sistema imunológico adaptativo. Há também, um número muito grande de componentes do sistema imunológico. Alguns são otimizados para atuarem contra um invasor específico enquanto outros são mais gerais para atuarem contra uma gama de invasores. O sistema imunológico é formado por redundâncias, ou seja, vários mecanismos de defesa atuam conjuntamente contra um único invasor.

Princípios Fundamentais e Elementos Constituintes

O sistema imunológico representa a principal barreira contra infecções, podendo responder de forma rápida e eficaz contra patógenos invasores. Além disso ele pode também realizar uma outra resposta mais lenta, porém eficaz e duradoura. Estas duas respostas são, respectivamente, a resposta inata e a resposta adaptativa.

Ambos os sistemas (inato e adaptativo) dependem da atividade das células brancas (leucócitos). A resposta inata se dá através, principalmente, dos macrófagos e granu-

lócitos, enquanto que a adaptativa se dá pelos linfócitos (principalmente células B e T), conforme está mostrado na figura 2.1 ¹. Os granulócitos se dividem em neutrófilos, eosinófilos e basófilos.

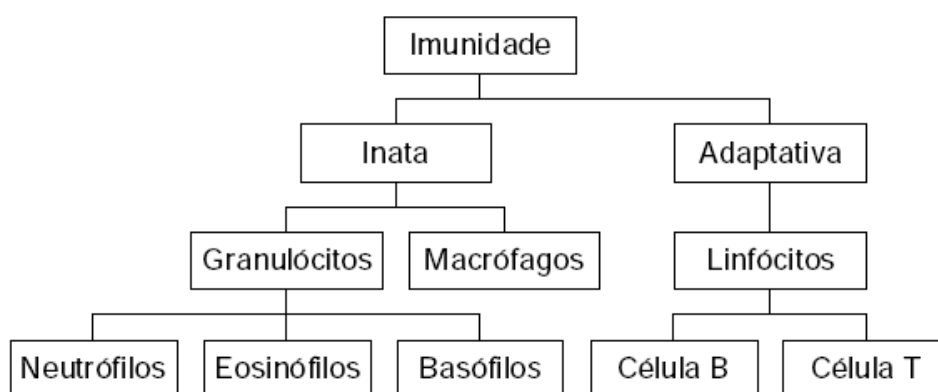


Figura 2.1: Mecanismos de defesa e seus principais mediadores.

Os componentes do sistema inato estão imediatamente disponíveis para o combate a um patógeno, sem exigir uma exposição anterior a este. Eles aparecem de modo semelhante em todos os indivíduos. Os macrófagos e neutrófilos possuem a capacidade de ingerir e digerir vários microorganismos e partículas antigênicas.

Uma resposta imune específica, ou seja, a produção de anticorpos dedicados para um determinado agente infeccioso é conhecida como resposta adaptativa. Os anticorpos são produzidos pelas células B em resposta às infecções, e sua presença em um indivíduo indica que este já foi vítima daquela infecção. Os linfócitos que atuam na resposta adaptativa são capazes de desenvolver uma memória imunológica. Eles reconhecem um mesmo antígeno, caso ele seja recorrente. Deste modo há uma resposta mais rápida do sistema adaptativo, evitando o restabelecimento da doença (de Castro, 2001). A resposta imune adaptativa melhora após cada exposição ao antígeno.

Desta maneira, a resposta imune adaptativa implica na imunidade contra a re-infecção de um determinado agente infectante, enquanto que a resposta inata permanece constante durante toda a vida de um indivíduo, independente da exposição ao antígeno.

¹As figuras 2.1 e 2.2 foram obtidas de de Castro (2001)

Em conjunto, os dois sistemas, inato e adaptativo, contribuem para uma defesa bastante eficaz, garantindo resistência a enfermidades mesmo em exposição constante aos mais diversos agentes patogênicos.

Sistema Imune Adaptativo

Todos os organismos vivos são capazes de apresentar alguma resistência a patógenos. A natureza desta resistência é distinta para cada tipo de organismo. Tradicionalmente, a imunologia aborda o sistema imunológico dos vertebrados, em especial, dos mamíferos (Klein, 1990). Os vertebrados desenvolveram, ao longo de milhares de anos, um sistema imunológico com a característica de ser preventivo. Os linfócitos são as principais células do sistema imune adaptativo. Cada linfócito virgem (linfócito que nunca encontrou seu antígeno específico) que penetra na corrente circulatória é portador de receptores de antígeno com uma única especificidade. A especificidade é determinada durante o desenvolvimento do linfócito por um mecanismo de rearranjo genético que tem a finalidade de gerar centenas de diferentes variantes dos genes codificadores de moléculas receptoras. Assim, embora cada linfócito seja capaz de reconhecer apenas um antígeno, a especificidade de cada linfócito é diferente dos demais e os milhões de linfócitos podem apresentar milhões de especificidades distintas. Os linfócitos sofrem um processo parecido com a seleção natural: somente aqueles que encontram um antígeno com o qual seu receptor pode interagir serão ativados para proliferar e se diferenciar em células memória ou em plasmócitos.

Pelo princípio da seleção clonal (isto será discutido posteriormente) pressupõe que uma grande quantidade de células B contendo receptores antigênicos, chamados anticorpos, estão circulando constantemente pelo nosso organismo. A grande variedade de células B é resultado de recombinação aleatória de fragmentos genéticos contidos no genoma, somada à inserção aleatória de seqüências genéticas conforme a célula se desenvolve. Esta grande variedade garante que pelo menos uma célula B irá produzir um anticorpo capaz de reconhecer um determinado antígeno (de Castro, 2001). A ligação

entre o anticorpo e o antígeno estimula a produção de clones desta célula, de modo que sucessivas gerações resultem no crescimento exponencial da quantidade de anticorpos do tipo selecionado circulando pelo corpo. Alguns destes anticorpos permanecem circulando mesmo após a eliminação do antígeno, constituindo a memória imunológica. Outras células se diferenciam em plasmócitos, produzindo anticorpos em grande escala. Finalmente, durante a reprodução, alguns clones sofrem o processo de maturação de afinidade, onde mutações são inseridas com altas taxas, o que melhora a capacidade destes anticorpos em reconhecerem os respectivos antígenos.

O funcionamento da resposta imune adaptativa como descrito, revela que o sistema imunológico pode ser visto como pertencente à teoria da evolução proposta por Charles Darwin, com três características fundamentais:

- diversidade da população: o sistema imunológico é capaz de produzir uma quantidade bem maior de anticorpos do que é utilizado instantaneamente. Aparentemente, grande parte dos anticorpos não executa nenhuma tarefa durante a resposta imunológica;
- variação genética: ocorre durante a produção de anticorpos através da recombinação de fragmentos genéticos das diversas bibliotecas do nosso genoma;
- seleção natural: anticorpos capazes de reconhecerem e se ligarem aos antígenos terão maiores probabilidades de se reproduzirem e serem mantidos como células de memória.

A similaridade entre a evolução biológica adaptativa e a produção de anticorpos ainda é mais forte quando consideramos que os dois processos centrais na produção de anticorpos, recombinação e mutação genética, são os mesmos responsáveis pela evolução biológica das espécies sexuadas.

Princípio da Seleção Clonal

Uma vez que cada célula possui seu receptor antigênico e que este é único, o número de linfócitos que pode se ligar a um determinado antígeno é limitado. A fim de produzir anticorpos específicos em quantidade suficiente para combater uma infecção, um linfócito ativado deve se proliferar a altas taxas. O princípio da seleção clonal está associado às características básicas da resposta imune adaptativa a um estímulo antigênico. Ele estabelece que apenas aquela célula que reconhece um estímulo antigênico irá se proliferar, sendo portanto, selecionada em relação às outras. Quando o corpo é exposto a um antígeno, uma subpopulação de linfócitos responde através da produção de anticorpos. Cada célula produz um único tipo de anticorpo, que é específico para o antígeno.

As células B além de se proliferarem e se diferenciarem em plasmócitos, também podem se diferenciar em células B de memória, caracterizadas por longo período de vida. As células de memória circulam pelo organismo e quando são expostas ao mesmo estímulo antigênico, começam a se diferenciar em plasmócitos e produzem anticorpos pré-selecionados pelo antígeno específico que estimulou a formação desta célula memória. Além disso, linfócitos em desenvolvimento e que se apresentam como potencialmente auto-reativos são removidos antes do amadurecimento.

A seleção clonal ocorre tanto nas células B quanto nas T. As células T não produzem anticorpos, mas são muito importantes na regulação da resposta das células B. De acordo com (de Castro, 2001) as principais características da teoria da seleção clonal são:

- eliminação ou inativação dos novos linfócitos diferenciados capazes de reagir com padrões antigênicos expressos por elementos do próprio organismo, denominados antígenos próprios. Esta característica assegura uma tolerância, ou ausência de resposta aos antígenos próprios;
- a interação de uma molécula estranha com um receptor de linfócito capaz de ligar a essa molécula leva à ativação linfocitária;

- restrição fenotípica de um padrão para cada célula diferenciada e retenção deste padrão pelos descendentes clonais;
- geração de variações genéticas aleatórias, através de um mecanismo de hipermutação somática, expressas sob a forma de diversos tipos de anticorpos.

Aprendizado por Reforço e Memória Imunológica

Para uma proteção eficiente de nosso organismo, somente o reconhecimento antigênico não é suficiente. É necessário, também, recursos suficientes para montar uma resposta imunológica efetiva contra agentes patogênicos. O número de clones específicos para cada antígeno é fundamental na determinação do resultado da infecção.

A aprendizagem imunológica envolve o aumento do tamanho da população e da afinidade, entre anticorpo e antígeno, de linfócitos que reconhecem um determinado antígeno. Como o número total de linfócitos é limitado, o aumento no tamanho de alguns clones implica na redução do número de outros.

Durante a evolução do sistema imunológico, o organismo encontra diversas vezes um mesmo antígeno. A resposta adaptativa inicial a um dado antígeno é composta por um conjunto pequeno de clones de células B, cada um produzindo anticorpos de diferentes especificidades (afinidades). A eficiência da resposta adaptativa a encontros posteriores pode ser consideravelmente aumentada através do armazenamento de células memória de grande afinidade com aquele antígeno. Deste modo tem-se um grande número de clones iniciais nos encontros subsequentes. Esta estratégia garante a velocidade e eficácia da resposta adaptativa. Este esquema é característico do aprendizado por reforço (Sutton and Barto, 1998), onde o sistema está continuamente melhorando sua capacidade de resposta. A figura 2.2 ilustra a memória imunológica. Considere que um antígeno Ag_1 é produzido no tempo 0. Poucos anticorpos específicos para Ag_1 estão presentes no sistema imunológico. Após uma fase de latência, os anticorpos para o antígeno começam a aumentar de concentração e afinidade até um certo patamar. Logo após a eliminação da infecção a concentração começa a cair. Quando outro antígeno

Ag_2 diferente é introduzido, o mesmo padrão de resposta é observado, deixando claro que os anticorpos de agora são distintos dos anticorpos para o antígeno Ag_1 . Isto demonstra a especificidade da resposta adaptativa. Uma outra característica importante da memória imunológica é sua capacidade de associação: células memória adaptadas a um antígeno Ag_1 apresentam uma resposta secundária mais rápida e eficiente para o antígeno Ag_1 e também para antígenos estruturalmente relacionados a ele, como o Ag'_1 . Este fenômeno é chamado de reação imunológica cruzada ou resposta reativa cruzada (de Castro, 2001).

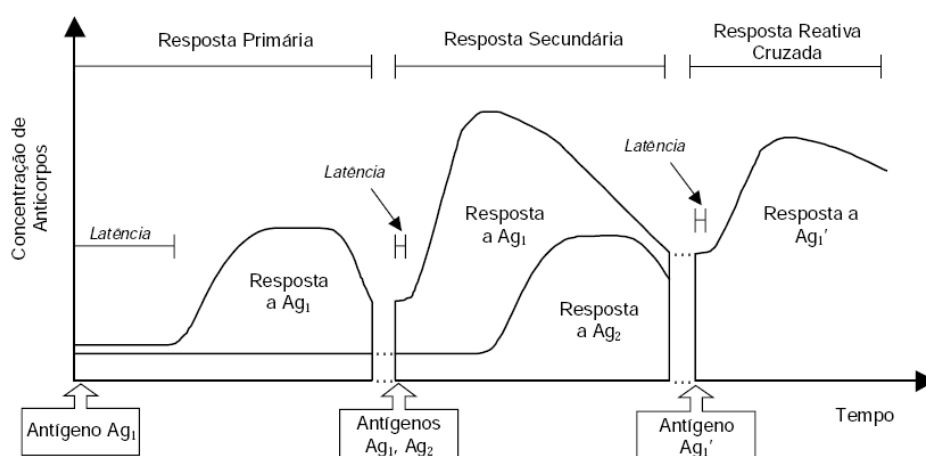


Figura 2.2: Resposta imunológica primária, secundária e reativa cruzada. Depois que um antígeno Ag_1 é visto uma vez (resposta primária), encontros subsequentes com o mesmo antígeno, ou um antígeno similar Ag'_1 (reação cruzada), provocará uma resposta mais rápida e eficiente (resposta secundária), não apenas a Ag_1 , mas também a Ag'_1 . As respostas primárias a antígenos distintos (Ag_1 e Ag_2) são similares.

Comparando-se a resposta primária com a secundária, a última é caracterizada por uma fase de latência menor e uma produção de anticorpos com uma taxa de produção maior. Uma dose de antígenos menor que a necessária para desencadear uma resposta primária pode levar a uma resposta secundária.

Maturação de Afinidade

Em uma resposta adaptativa auxiliada pelas células T, o repertório de células B ativadas por antígenos é diversificado basicamente por dois mecanismos: hipermutação somática e edição de receptores. Somente os descendentes com alta afinidade antigênica são selecionados para fazerem parte das células de memória.

Os anticorpos que possuem maior afinidade em uma resposta secundária possuem, na média, maiores afinidades do que aqueles das respostas primárias. Este fenômeno é chamado de maturação de afinidade, e é restrito às respostas dependentes das células T. A maturação requer que a ligação dos linfócitos com o antígeno, na resposta secundária, seja estruturalmente diferente daquelas apresentadas na resposta primária.

Mutações aleatórias ocorridas durante a expansão clonal dos linfócitos B podem levar a um aumento da afinidade do anticorpo ao estímulo antigênico que o selecionou. Esses descendentes com alta afinidade antigênica são selecionados para fazerem parte das células de memória. Assim, um mecanismo deve existir para que células B com alta afinidade sejam selecionadas e predominem nas respostas futuras. Devido à natureza aleatória, uma grande parcela das células B mutantes tornam-se não-funcionais, ou desenvolvem-se receptores auto-reativos, que reagem ao próprio organismo. As células com baixa afinidade, ou as células auto-reativas, devem ser eficientemente eliminadas, de forma que não contribuam com as células de memória.

O desenvolvimento de anticorpos demonstra o papel da mutação, seguida da seleção na maturação da resposta imunológica adaptativa. O aumento da afinidade da resposta primária para a secundária, e assim sucessivamente, demonstra que a maturação é um processo contínuo.

Existem três aspectos fundamentais de uma resposta imune adaptativa: a diversidade suficiente para combater um universo de antígenos, distinção entre próprio e não-próprio e memória imunológica. Na teoria da seleção clonal, a memória seria fornecida pela expansão do número de um clone específico e a mutação aleatória seguida de seleção permitiria o aumento da afinidade deste clone. Ocasionalmente alguns linfócitos B

sofrem edição de receptores, ou seja, células auto-reativas são eliminadas e desenvolvem-se novos receptores através de recombinação genética. A teoria de seleção clonal pode acomodar a edição de receptores caso esta ocorra antes da seleção celular. Qualquer clone com alta afinidade produzido por hipermutação somática ou edição de receptores seria preferencialmente mantido, mas células com baixa afinidade também poderiam fazer parte do repertório mantendo a diversidade populacional.

Regulação do Mecanismo de Hipermutação

Um rápido acúmulo de mutações é necessário para uma rápida maturação da resposta imunológica. No entanto, a maioria das mudanças introduzidas resultará em anticorpos auto-reativos ou de baixa afinidade. Uma célula que sofreu mutação e foi capaz de aumentar sua afinidade continua a sofrer mutação com a mesma taxa nas próximas respostas imunológicas pode estar sujeita a uma perda das variações que levaram à melhoria da afinidade. Assim, um curto pico de hipermutação, seguido de um intervalo para seleção e expansão clonal devem formar a base do processo de maturação. A seleção deve fornecer uma regulação do processo de hipermutação, que é dependente da afinidade. Células com baixa afinidade permanecem sendo mutadas, enquanto células com alta afinidade devem ter suas taxas de mutação controladas, geralmente de maneira gradual, ou até inativadas (de Castro, 2001).

2.1.2 Estrutura Básica do Algoritmo de Seleção Clonal

O Algoritmo de Seleção Clonal, ou simplesmente Clonal, é baseado nos princípios do sistema imunológico que acabou de ser descrito. Uma importante propriedade do mesmo é sua capacidade de armazenar algumas soluções sub-ótimas juntamente com o ótimo global (de Castro and Von Zuben, 2002).

O princípio da seleção clonal, como mostrado na seção 2.1.1, é um processo evolucionário natural no sistema imunológico. Células que produzem anticorpos capazes de reconhecer um dado antígeno são selecionadas pra proliferarem através de clones. Os

clones são submetidos a um processo chamado de maturação de afinidade, descrito na seção 2.1.1, caracterizado pela hipermutação. A produção de clones é proporcional à afinidade da célula em produzir anticorpos para um dado antígeno, já a hipermutação é inversamente proporcional à afinidade. Assim, o processo de maturação de afinidade pode ser visto como um mecanismo de refinamento local. Ele promove o surgimento de anticorpos que melhor combatem o antígeno, isto é, que têm melhor afinidade antigênica. O sistema imunológico tem ainda um mecanismo de preservação de diversidade, através da edição de receptores, item 2.1.1. Se essas características fazem do algoritmo clonal uma boa ferramenta para resolver problemas de otimização, particularmente otimização multimodal. O mecanismo de memória imunológica preserva um conjunto de soluções ótimas, contendo o ótimo e vários sub-ótimos, do problema.

A analogia das variáveis do problema de otimização com os termos biológicos é então:

- antígeno: é o problema a ser otimizado (função objetivo);
- anticorpos: são as soluções candidatas para resolver o problema de otimização;
- afinidade: é o valor de *fitness* dos anticorpos.

Baseados no princípio da seleção clonal e nas considerações feitas acima, um algoritmo clonal pode ser sintetizado com os seguintes passos:

1. uma população de N anticorpos é gerada;
2. a afinidade de cada anticorpo é avaliada por meio da função objetivo;
3. o princípio da seleção clonal é aplicado. Os anticorpos com melhor afinidade geram mais clones;
4. aplicação da hipermutação aos clones. Os anticorpos com baixa afinidade têm uma taxa maior de mutação;

5. aplicação à edição de receptores. Os piores indivíduos são trocados por outros gerados aleatoriamente, para manter a diversidade;
6. os anticorpos da população corrente são armazenados na população *off-line*. É então aplicada a supressão nesta população para eliminar anticorpos que têm alta afinidade entre si;
7. retorna ao passo 2 até que um critério de parada seja atendido.

No próximo capítulo serão discutidos em mais detalhes o algoritmo clonal e os algoritmos meméticos aplicados ao projeto de redes de distribuição de energia elétrica.

2.2 Algoritmos Meméticos

2.2.1 Descrição

Os Algoritmos Evolucionários (AE) têm funcionamento bastante simples e com regras bastante fáceis de serem implementadas computacionalmente. Por isso, esses algoritmos têm sido amplamente aplicados nos mais diversos problemas de otimização. Os Algoritmos Genéticos (Goldberg, 1989), Algoritmos de Colônia de Formigas - *Ant Colony Algorithms* (Dorigo et al., 1996; Dorigo and Gambardella, 1997; Dorigo et al., 1999) e Sistemas Imunológicos Artificiais (de Castro and Timmis, 2003; de Castro and Von Zuben, 2002; de Castro, 2001) são os algoritmos evolucionários mais conhecidos e utilizados em otimização. Os trabalhos (Ramírez-Rosado and Bernal-Agustín, 1998; Carvalho et al., 1998; Ching-Tzong and Guor-Rurng, 2002; Gómez et al., 2004; Carrano et al., 2006, 2007a) exemplificam a utilização de algoritmos evolucionários aplicados à problemas de redes de distribuição.

Em geral, podemos citar as características de exploração do espaço de busca, tanto de aspecto global como local, que fazem destes algoritmos ferramentas de busca global robustas e efetivas quando se deseja encontrar um mínimo global aproximado (Wanner,

2006). A natureza exploratória desses algoritmos permite encontrar regiões do espaço de busca onde se encontram soluções muito boas. Por outro lado, ela prejudica a convergência e precisão nos estágios finais do processo de busca. Nesses estágios finais, pouca informação nova é incorporada através dos mecanismos de busca local ao passo que os mecanismos de busca global introduzem informações com perturbações muito fortes, impossibilitando a convergência com alta precisão (Jong, 1993; Mitchell, 1996; Goldberg and Voessner, 1999).

Com o intuito de tornar os AEs ferramentas que mantenham a capacidade de fazer a busca global por todo o espaço de busca e, ainda, permitam ao algoritmo encontrar soluções mais precisas e robustas surgiram os algoritmos híbridos. Existe uma variedade de classificação muito grande para os algoritmos e eles são chamados de Algoritmos Meméticos, Algoritmos Lamarkianos, Algoritmos Baldwinianos, Algoritmos Culturais, Busca Local Genética, etc. (Goldberg and Voessner, 1999; Davis, 1991). Neste texto será utilizada a denominação de Algoritmos Meméticos, por englobar as demais classificações (Moscato, 1989; W. Hart and Smith, 2005; J. Smith and Krasnogor, 2004). Um detalhamento mais minucioso da taxonomia desses tipos de algoritmo está em (Krasnogor and Smith, 2005).

Algoritmos Meméticos (AM) são então algoritmos evolucionários combinados com métodos de busca local com o intuito de propiciar resultados mais refinados (Moscato, 1989). Eles são baseados na inspiração biológica do algoritmo utilizado e na noção de memes de Dawkins, em que um meme é uma unidade de evolução cultural capaz de permitir refinamentos locais. No contexto de otimização heurística, um meme representa uma estratégia de aprendizado, incorporando ao indivíduo o equivalente à sua evolução cultural durante seu período de vida, tornando possível capturar informações que um modelo puramente evolutivo não seria hábil para tal (Krasnogor and Smith, 2005). Algoritmos Meméticos podem ser considerados como um casamento entre a habilidade de busca global consagrada dos algoritmos evolutivos e a melhoria obtida pelo mecanismo de busca local. Isto permite associar as vantagens de exploração dos AEs (generalidade,

robustez, busca global eficiente) aos mecanismos de busca local (exploração aplicada ao problema específico, convergência para mínimos locais). Essas características dos AMs levaram a resultados importantes em diversos problemas, superando os resultados obtidos por meio dos AEs. Exemplos de aplicações bem sucedidas de Algoritmos Meméticos podem ser vistos tanto em problemas contínuos (Wanner, 2006) quanto em problemas combinatórios. A seguir listamos uma série de problemas combinatórios típicos que podem ser resolvidos com o uso dos AM's: problema do caixeiro viajante (*Traveling Salesman Problem - TSP*) (Freisleben and Merz, 1996; H. M. Voigt and Schwefel, 1996; Merz and Freisleben, 1997; Nagata and Kobayashi, 1997; J. Watson and Howe, 1998; Krasnogor and Smith, 2000); *Quadratic Assignment Problem - QAP* (Vazquez and Whitley, 2000; Even, 1993; Merz and Freisleben, 1999a; J. Tang and Ong, 2006); *Binary Quadratic Programming - BQP* (Merz and Freisleben, 1999b); *Minimum Graph Coloring* (MGC) (D. Costa and Dubouis, 1995; Fleurent and Ferland, 1997; Dasgupta et al., 1998) e *Protein Structure Prediction Problem* (PSP) (N. Krasnogor and Hirst, 2002; R. Carr and Smith, 2002).

A literatura trata como principais aspectos a serem considerados em um algoritmo memético a periodicidade de aplicação da busca local, a escolha das soluções sobre as quais devem ser aplicadas a busca local, a intensidade da busca local, o método de busca local a ser utilizado e a definição da vizinhança sobre a qual será efetuada a busca local. Essas questões têm como fundamento proporcionar um equilíbrio entre a busca local e a busca global a fim de melhorar o desempenho do AM (Krasnogor and Smith, 2005). A definição de vizinho é, também, de fundamental importância na busca local, assim como o meio de obter os vizinhos.

As respostas para essas questões são específicas para cada tipo de problema e dependem da estrutura do algoritmo memético empregado. Não existem ainda respostas bem definidas que garantem a qualidade do AM. O que se faz é tentar adaptar o algoritmo ao problema em questão.

Capítulo 3

Algoritmos Utilizados no Projeto de Redes de Distribuição de Energia Elétrica

3.1 Algoritmo Clonal para o Projeto de Redes de Distribuição de Energia Elétrica

O algoritmo clonal utilizado para o projeto de redes de distribuição de energia elétrica utilizado neste trabalho segue os seguintes princípios: os anticorpos (soluções candidatas) com maior afinidade antigênica (com melhor avaliação da função objetivo) têm grande taxa de proliferação (gera maior número de clones). Esses clones são submetidos ao processo de maturação de afinidade (mutação com taxa inversamente proporcional à afinidade). O raio da mutação (a severidade do processo de mutação aplicado ao clone) é inversamente proporcional à afinidade. Assim, para os melhores anticorpos (melhores soluções), seus clones sofrem uma mutação que tem maior probabilidade de gerar anticorpos bem próximos ao anticorpo de origem, habilitando assim uma busca local. Para os anticorpos pouco adaptados (piores soluções) a mutação aplicada é mais “severa”, fazendo com que os anticorpos obtidos sejam distantes do anticorpo de origem, habilitando uma busca global. Portanto, o mecanismo de maturação é visto como uma busca global com mecanismos de refinamento local, que faz com que

os anticorpos ataquem o antígeno mais eficazmente (minimize a função objetivo).

Uma população *off-line* foi incorporada ao algoritmo clonal para aumentar a capacidade deste algoritmo de mapear as soluções sub-ótimas (esta população é normalmente empregada no algoritmo de rede imunológica (de Castro and Timmis, 2002)). Os anticorpos desta população são comparados entre si para verificar a afinidade entre eles. Caso haja anticorpos que tenham uma afinidade maior que um dado limite, o anticorpo que pior responde ao antígeno é eliminado da população. Para avaliar a afinidade entre anticorpos utilizamos a métrica *T-norm*. Assim, anticorpos que tenham uma distância menor que um dado limite σ , têm afinidade muito grande e um dos dois deve ser eliminado. Este processo de eliminação de anticorpos que têm alta afinidade com outros anticorpos é chamado de supressão.

O algoritmo utilizado para a geração da população inicial é o algoritmo Prune (algoritmo de Kruskal inverso) considerando todas as conexões possíveis e depois apenas conexões escolhidas aleatoriamente. Para o passo seguinte do algoritmo, é necessário definir um parâmetro B ($0 \leq B \leq 1$) que determine qual a fração dos anticorpos da população corrente será selecionada para o processo de maturação de afinidade. A afinidade antigênica de cada anticorpo é medida pela avaliação da função objetivo (equação (1.16)), e todas as soluções são ordenadas em ordem crescente. As $B \cdot N$ melhores soluções, onde N é o número de soluções, são selecionadas para o processo de maturação de afinidade. O número de clones gerados para cada anticorpo é dado por (3.1):

$$NC_i = \text{round} \left(\frac{\beta \cdot N}{i} \right), \quad i = 1, \dots, \lceil B \cdot N \rceil \quad (3.1)$$

onde β é uma constante ($0 \leq \beta \leq 1$). Como um exemplo, para $B = 0,5$, $\beta = 1$ e $N = 50$, o melhor anticorpo ($i = 1$) irá dar origem à 50 clones ($NC_1 = 50$), o segundo melhor ($i = 2$) à 25 ($NC_2 = 25$) e assim por diante até o 25º melhor anticorpo.

Uma vez determinados os números de clones para cada anticorpo, cada um desses clones passa pelo processo de maturação de afinidade. Para a determinação do raio

de mutação existe um parâmetro auxiliar r_b , chamado de raio base de mutação, que tem a função de dimensionar o raio de mutação através da seguinte equação: $\alpha = r_b \cdot e^{-fit_{iteracao-1}}$, onde $fit_{iteracao-1}$ é o valor da *fitness* na iteração anterior. Como esta equação é uma exponencial com o expoente negativo, o raio de mutação é inversamente proporcional à *fitness* do indivíduo, sendo seu valor regulado pelo raio base de mutação. Existem anticorpos, que devido à escolha do parâmetro B , não são selecionados para dar origem à clones. Esses anticorpos são substituídos por novos anticorpos gerados aleatoriamente, como uma medida de manter a diversidade da população.

Diferentemente do algoritmo genético, o algoritmo clonal não possui um mecanismo de cruzamento entre soluções. Elas são mantidas dispersas sobre o espaço de busca, não sendo forçadas a se concentrarem em uma determinada região. Por isso, o algoritmo clonal é capaz de mapear soluções que tendem a se concentrar em regiões de mínimos locais do espaço de busca, obtendo um conjunto de soluções, a solução ótima e soluções sub-ótimas. Há ainda a vantagem de sua simplicidade de implementação e a necessidade de se fixar poucos parâmetros, que são: N , B , β e r_b . A estrutura do algoritmo clonal implementado para o projeto de redes de distribuição de energia elétrica utilizado neste trabalho é apresentada abaixo.

Estrutura Básica do Algoritmo de Seleção Clonal

```

pop0 ← gerar população inicial;
fob0 ← avalia funcao objetivo(pop0);
fit0 ← calcula fitness(pop0, fob0);
popext ← popo;
conv ← false;
iteracao ← 1;
while conv ← false do
    (popiteracao-1, fobiteracao-1) ← ordenar(popiteracao-1, fobiteracao-1)
    for i = 1 até (B · N) do
        popiteracaoi ← popiteracao-1i;
        fobiteracaoi ← fobiteracao-1i;
        NCi ← round( $\frac{\beta \cdot N}{i}$ );

```

```

for j= 1 até  $N_C^i$  do
   $cl \leftarrow \text{clone}(pop_{iteracao}^i)$ ;
   $\alpha = r_b \cdot e^{-fit_{iteracao-1}^i}$ ;
   $cl \leftarrow \text{mutacao}(cl, \alpha)$ ;
   $fob_{cl} \leftarrow \text{avalilar}(cl)$ ;
  if  $fob_{cl} < fob_{iteracao}^i$  then
     $pop_{iteracao}^i \leftarrow cl$ ;
     $fob_{iteracao}^i \leftarrow fob_{cl}$ ;
  end if
end for
end for
for i = round( $B \cdot N$ ) até N do
   $pop_{iteracao}^i \leftarrow \text{gerar anticorpo}$ ;
   $fob_{iteracao}^i \leftarrow \text{avalia}(pop_{iteracao}^i)$ ;
end for
if  $\text{resto}(\frac{iteracao}{atualizapop_{ext}}) = 0$  then
   $pop_{ext} \leftarrow \text{agrupa}(pop_{ext}, pop_{iteracao})$ ;
   $pop_{ext} \leftarrow \text{supressao}(pop_{ext})$ ;
end if
end while

```

A seguir alguns aspectos importantes do algoritmo são descritos:

- Avaliação da população e atribuição da fitness: os indivíduos da população são avaliados através da função objetivo do problema. A *fitness* tem o papel de, juntamente com o fator r_b , controlar o raio de mutação do algoritmo. Após a ordenação da população pela avaliação da função objetivo, a *fitness* dos indivíduos é calculada por 3.2.

$$fit^i = 1 - \frac{r}{N - 1} \quad (3.2)$$

onde r é um número que define o *rank* do indivíduo i , variando de 0 (melhor indivíduo) à $N - 1$ (pior indivíduo);

- Operador de mutação: este operador de mutação é uma aplicação direta da propriedade de gerar redes aleatórias à distâncias pré-definidas com o uso da métrica *T-norm*. Dada uma rede pai N_o e um raio de mutação $r = r_b \cdot e^{fit}$, o operador de mutação obtém uma rede N_o^k , tal que a distância entre N_o e N_o^k seja próxima de r . Para gerar a rede N_o^k , o operador de mutação segue uma sequência de passos conforme apresentado na seção 1.4.2;
- Convergência: pelo fato de ser imposta no algoritmo nenhuma pressão, as soluções ficam dispersas pelo espaço de busca. Não há uma concentração das soluções ao redor da melhor solução. O algoritmo clonal apresenta, portanto, uma convergência mais lenta que o algoritmo genético. O critério de convergência mais utilizado é o número máximo de gerações. Pode ser utilizado também um critério de estabilização.

As restrições para o projeto de redes de distribuição de energia, apresentadas na Sec. 1.5, são tratadas da seguinte maneira:

- Conectividade e radiabilidade: a geração da população inicial e o operador de mutação são construídos de tal forma que sempre as redes resultantes são factíveis (redes cujo grafo é uma árvore). Portanto, estas restrições não são violadas por construção do algoritmo;
- Capacidade de condução de corrente nas conexões: se o tipo de cabo utilizado não é capaz de conduzir a corrente que deve passar na conexão, este cabo é trocado por um de capacidade imediatamente superior. Caso não seja mais possível alterar o tipo de cabo (a conexão já está com o último tipo de cabo disponível), a afinidade da solução recebe um fator de penalização;
- O nível de tensão em cada nó é obtido pelo algoritmo de fluxo de potência após a fixação dos tipos de cabo para cada conexão. Caso a tensão em pelo menos um nó viole os limites de tensão, a afinidade desta solução é penalizada por um fator.

É importante notar que as penalizações sobre as soluções somente é empregada durante o processo de otimização. Ao fim da execução do algoritmo de otimização, todas as soluções obtidas são soluções factíveis para o cenário sob o qual se faz a otimização. Durante a análise de sensibilidade multiobjetivo, que será vista a seguir, as soluções que violam pelo menos uma das restrições são consideradas infactíveis, mesmo que uma mudança de tipo de conexão a torne factível.

3.2 Definição de Vizinhaça

Para a definição de vizinhaça são utilizados os conceitos apresentados na seção 1.4 que descrevem como se apropriar de propriedades do espaço contínuo no espaço de redes. A métrica proposta permite caracterizar geometricamente o espaço de redes e definir aspectos importantes, tais como: posição relativa entre redes, caracterização dos conjuntos factíveis dentro do espaço de redes, segmentação do espaço de busca e direções no espaço de redes.

Serão consideradas redes vizinhas de uma determinada rede aquelas redes que estiverem a distâncias menores ou iguais a um valor pré-estabelecido. A este valor pré-estabelecido denomina-se *Raio Local* (RL). Desta forma:

$$\vec{N}_i \in V \Leftrightarrow \left| \vec{N}_s - \vec{N}_i \right| \leq RL \quad (3.3)$$

onde:

\vec{N}_s é a rede que se deseja obter a vizinhaça;

V é a vizinhaça da rede N .

A definição do valor de RL é feita por meio do mapeamento de todos os nós que são folhas do grafo em árvore desta rede. Folhas são aqueles nós extremos da rede onde só existe uma única conexão. A seguir é dado um exemplo que ilustra o cálculo de RL para a rede da figura 3.1

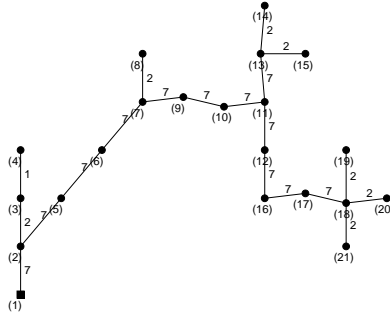


Figura 3.1: Exemplo de rede para calcular RL

As folhas da rede da figura 3.1 são os nós 4, 8, 14, 15, 19, 20 e 21. O nó 1 não é considerado folha. Ele é a raiz da rede que é uma sub-estação de onde é distribuída toda a energia para a rede.

É então feita a desconexão de uma folha de cada vez, não sendo permitida a desconexão concomitante de duas ou mais folhas em uma mesma rede. Após desconectado um dado nó, são verificadas as possibilidades de reconexão deste nó com base na codificação adotada, sendo obtida uma rede N_s^k para cada reconexão possível, (k o número de possíveis reconexões). Cada reconexão pode ser realizada por t_c diferentes tipos de cabos, existem portanto $k \cdot t_c$ redes diferentes que podem ser obtidas pela reconexão de uma única folha. Fazendo isto para todas as folhas da rede, chega-se a $l \cdot k \cdot t_c$ redes obtidas, onde l é o número de folhas. Cada vez que se desconecta uma folha, $k \cdot t_c$ redes derivadas desta desconexão são criadas. O procedimento é então repetido para todas as folhas. São então medidas as distâncias entre a rede N_s e as $l \cdot k \cdot t_c$ redes obtidas, e é tomada como *Raio Local* a maior destas distâncias.

A justificativa para esta metodologia de obtenção de RL é que consideram-se como redes locais a uma rede dada todas aquelas redes que estão distantes desta por uma distância máxima. Esta distância máxima corresponde a uma mudança nas extremidades da rede sem que haja uma mudança considerável em suas características. São ainda possíveis mudanças nos tipos de cabos das conexões. As redes locais assim definidas apresentam valores baixos de distância medidas pela métrica T -norm. Isto parece co-

erente com a noção intuitiva de localidade.

Abaixo é apresentado um pseudo-código para a obtenção do *Raio Local*. Este algoritmo segue exatamente os passos que acabaram de ser descritos.

Estrutura Básica do Algoritmo para a Raio Local

```

RL ← infinito;
FOLHAS ← localizafolhas(Ns);
NFOLHAS ← tamanho(FOLHAS);
for i = 1 até NFOLHAS do
    Nsi ← Ns;
    Nsi(FOLHAS(i)) ← 0;
    RECONEXOES ← localizaconexoes(Nsi);
    NRECONEXOES ← tamanho(RECONEXOES);
    for k = 1 até NRECONEXOES do
        for j = 1 até tc do
            Nsi(RECONEXOES(k)) ← j;
            DISTANCIA ← metrica(Ns, Nsi);
            if DISTANCIA < RL then
                RL ← DISTANCIA;
            end if
        end for
    end for
end for

```

Onde:

N_s é a rede para a qual se deseja calcular o *Raio Local* para definir sua vizinhança;

RL é o valor do *Raio Local* para a rede *N_s*;

FOLHAS são os nós da rede *N_s* que representam as folhas do grafo em árvore desta rede;

localizafolhas é a sub-rotina que localiza dentre todos os nós de uma rede quais são folhas, dado que a rede é representada por um grafo em árvore;

NFOLHAS é o número de folhas encontradas;

N_s^i é a rede após sofrer a desconexão das folhas e a reconexão;

RECONEXOES são todas as possibilidades de reconectar a folha após sua desconexão;

localizareconexoes é a sub-rotina que localiza todas as possíveis reconexões para a folha, dada a codificação usada;

NRECONEXOES é o número de possibilidades de reconectar o nó da rede que representa a folha;

DISTANCIA é a distância obtida via métrica *T-norm* entre as redes N_s e N_s^i ;

metrica é a sub-rotina que computa a distância entre duas redes através da métrica *T-norm*.

3.3 Método de Busca Local

Além de não serem variados os métodos de busca local disponíveis na literatura para problemas combinatórios, ainda existe a restrição de estes métodos serem muitas vezes exclusivos para o problema em questão e para a codificação de variáveis utilizada (Ong and Keane, 2004). Por exemplo, o método de busca local *2-opt algorithm*, para o Problema do Caixeiro Viajante (Merz and Freisleben, 1997; J. Watson and Howe, 1998) não é um bom método a ser implementado no projeto de redes, uma vez que a maioria das soluções obtidas seriam redes inactivas. Essas questões motivaram a implementação de métodos de busca local específicos para o projeto de redes de distribuição de energia elétrica.

A primeira inspiração que se tem para implementar a busca local em problemas combinatórios é a utilização da métrica de Hamming, na qual variam-se todas as possibilidades de conexões e de ramos para uma dada rede, uma de cada vez. Esta é uma definição clássica para o conceito de vizinhança em problemas combinatórios. Com o uso desta metodologia existe um número muito grande de possibilidades de redes vizinhas. Este número cresce à medida que cresce a codificação da rede. Para uma rede de n vari-

áveis e t_c tipos de conexões possíveis para cada variável, teríamos $n \cdot (n - 2) \cdot t_c + (t_c - 1)$ redes vizinhas. Seria, portanto, necessário avaliar as $n \cdot (n - 2) \cdot t_c + (t_c - 1)$ redes para se saber qual rede é mínimo local. Esta quantidade de avaliações torna esta metodologia inviável do ponto de vista de tempo computacional. Ainda mais, uma vez que são permitidas perturbações em todas as variáveis, podem ser levadas em consideração algumas redes que têm características completamente diferentes das redes para a qual se faz a busca local. Essas redes muito diferentes não deveriam ser consideradas redes vizinhas. Além do problema de descontinuidade do espaço de busca causado pela utilização da métrica de Hamming, conforme demonstrado em Carrano (2007).

Neste trabalho, todos os métodos de busca local foram implementados com base na definição de vizinhança feita na seção 3.2. A adoção do *Raio Local* como medida de vizinhança, assegura às redes características semelhantes, permitindo assim sua classificação como sendo locais. Estes métodos têm ainda uma característica “gulosa”. Quando se obtém uma rede vizinha e que é melhor que a rede dada, há uma atualização imediata da rede para a qual se está fazendo a busca local.

O primeiro método de busca local implementado é baseado na geração de redes aleatórias à distâncias pré-definidas no espaço de redes. As distâncias são calculadas usando a métrica *T-norm*. Esta é também a maneira como o operador de mutação do algoritmo é construído.

A busca local será feita para distâncias menores ou iguais à *RL*. Este método será chamado de *Método de Busca Local 1 (MBL1)*.

Estrutura Básica do Algoritmo para o Método de Busca Local 1

```

cont ← 0;
CRITERIO ← 1;
while CRITERIO do
  cont ← cont + 1;
  RL ← distancialocal( $N_s$ );
  dist ← rand · RL;
   $N_{si}$  ← rededist( $N_s$ , dist);

```

```

 $FN_{si} \leftarrow \text{avalia}(N_{si});$ 
if  $FN_{si} < FN_s$  then
     $FN_s \leftarrow FN_{si};$ 
     $N_s \leftarrow N_{si};$ 
     $cont \leftarrow 0;$ 
end if
if  $cont > \text{maxcont}$  then
     $CRITERIO \leftarrow 0;$ 
end if
end while

```

Onde:

N_s é a rede sobre a qual se deseja fazer a busca local;

$cont$ variável que acumula o somatório do número de redes vizinhas à N_s que foram geradas;

$CRITERIO$ é o critério de parada do algoritmo, enquanto $CRITERIO$ for 1 o algoritmo está em execução;

$dist$ é a distância sobre a qual será criada um rede vizinha à N_s ;

$rand$ é um número aleatório entre 0 e 1;

N_{si} é a rede vizinha a N_s , distante desta de uma distância $dist$;

rededist sub-rotina que realiza o processo de caminhar a rede N_{si} em direção à N_s até ficar à distância $dist$;

FN_{si} é a avaliação da rede N_{si} ;

avalia faz a avaliação da rede sobre a função objetivo;

FN_s é a avaliação da função objetivo para a rede N_s ;

$maxcont$ é o número máximo de redes que podem ser avaliadas neste método de busca local;

RL é a *Raio Local* para a rede N_s ;

distancialocal sub-rotina que calcula o valor de RL para a rede N_s .

O segundo método, *Método de Busca Local 2 (MBL2)*, é uma derivação do primeiro, sendo que a diferença entre eles é a intensidade da busca local realizada. Aqui, são geradas N_{redes} redes locais através do princípio de funcionamento do *MBL1*. Dentre essas N_{redes} redes e a rede inicial, escolhe-se a melhor rede como sendo o resultado da busca local. Este método é muito barato computacionalmente comparado ao *MBL1*. Ele não apresenta uma característica gulosa e baseá-se na avaliação e no ordenamento de algumas soluções e a escolha *a posteriori* do mínimo local.

Estrutura Básica do Algoritmo para o Método de Busca Local 2

```

cont ← 1;
for cont = 1 até  $N_{redes}$  do
   $RL \leftarrow \text{distancialocal}(N_s)$ ;
   $dist \leftarrow rand \cdot RL$ ;
   $N_{si}(cont) \leftarrow \text{rededist}(N_s, dist)$ ;
   $FN_{si}(cont) \leftarrow \text{avalia}(N_{si})$ ;
  cont ← cont + 1;
end for
 $FN_s \leftarrow \text{min}(FN_s, FN_{si})$ ;
 $N_s \leftarrow \text{posmin}(N_s, N_{si})$ ;

```

Onde:

N_s é a rede sobre a qual se deseja fazer a busca local;

$cont$ indica o índice sobre o qual será armazenada a rede obtida e a avaliação;

N_{redes} é o número de redes que se quer que sejam geradas no processo de busca local;

$dist$ é a distância sobre a qual será criada um rede vizinha à N_s ;

$rand$ variável aleatória entre 0 e 1;

N_{si} é a rede vizinha a N_s , distante desta de uma distância $dist$;

rededist sub-rotina que realiza o processo de caminhar a rede N_{si} em direção à N_s até ficar à distância $dist$;

FN_{si} avaliação da rede N_{si} ;

avalia faz a avaliação da rede sobre a função objetivo;

FN_s é a avaliação da função objetivo para a rede N_s ;

min indica qual é o menor valor dentre os dados apresentados;

RL é a *Raio Local* para a rede N_s ;

distancialocal sub-rotina que calcula o valor de RL para a rede N_s ;

posmin sub-rotina que retorna qual rede, entre as redes N_s e N_{si} que possui a menor avaliação para a função objetivo.

3.4 Algoritmos Meméticos Adaptados ao Projeto de Redes de Distribuição

Neste trabalho, os algoritmos meméticos aplicados ao projeto de redes de distribuição de energia elétrica são uma hibridação entre o Algoritmo Clonal e um método de busca local.

A escolha pelo algoritmo clonal se justifica pela sua aplicação, com bons resultados, no projeto de redes de distribuição de energia elétrica com incertezas na evolução carga, como pode ser visto na referência (Carrano et al., 2007a). O Algoritmo Clonal foi utilizado para obter redes ótimas em projeto de redes a longo prazo, onde incertezas inerentes à este tipo de projeto foram consideradas.

No entanto, os algoritmos clonais, como a maioria dos algoritmos evolucionários, não oferecem resultados muito refinados. Muitas vezes nem mesmo um número muito grande de execuções destes algoritmos permite obter ótimos exatos para um dado problema. É com o intuito de amenizar esta limitação do algoritmo que foi implementado o algoritmo memético: clonal-busca local. O papel do método de busca local é refinar as soluções obtidas pelo método clonal. Desta forma, espera-se a obtenção de resultados mais refinados para o projeto de redes de distribuição de energia elétrica com incertezas na evolução da carga. Espera-se, ainda, um melhor mapeamento dos sub-ótimos.

As principais questões que serão consideradas na implementação dos algoritmos

meméticos são a definição de vizinhança a ser utilizada na busca local e o método de busca local específico para o projeto de redes de distribuição de energia elétrica. Foram então construídos três algoritmos meméticos, que são apresentados a seguir:

Algoritmo Memético 1 (AM1): o método de busca local *MBL1* foi associado ao algoritmo clonal para a construção do *AM1*. O *MBL1* é empregado em todas as soluções geradas ao fim da execução do algoritmo clonal. A população considerada é a população *off-line*. A estrutura deste algoritmo é mostrada em pseudo código a seguir.

Algoritmo Memético 2 (AM2): o método de busca local *MBL2* foi associado ao algoritmo clonal para a construção do *AM2*. O *MBL2* é empregado em intervalos de k em k gerações usando a população atual do algoritmo clonal. O pseudo código mostrado a seguir contém os passos relativos ao *AM2*.

Algoritmo Memético 3 (AM3): os métodos de busca local *MBL1* e *MBL2* são associados ao algoritmo clonal para a construção do *AM3*. O *MBL2* é empregado em intervalos de k em k gerações usando a população atual do algoritmo clonal. O método *MBL1* é empregado sobre todas as soluções, usando a população *off-line*. O pseudo código mostrado a seguir contém os passos relativos ao *AM3*.

Estrutura Básica do Algoritmo Memético

```

pop0 ← gerar população inicial;
fob0 ← avalia funcao objetivo(pop0);
fit0 ← calcula fitness(pop0, fob0);
popext ← pop0;
conv ← false;
iteracao ← 1;
while conv ← false do
    (popiteracao-1, fobiteracao-1) ← ordenar(popiteracao-1, fobiteracao-1)
    for i = 1 até (B · N) do
        popiteracaoi ← popiteracao-1i;
        fobiteracaoi ← fobiteracao-1i;
        NCi ← round( $\frac{\beta \cdot N}{i}$ );
        for j= 1 até NCi do

```

```

     $cl \leftarrow \text{clone}(pop_{iteracao}^i);$ 
     $\alpha = r_b \cdot e^{-f_{it_{iteracao-1}}^i};$ 
     $cl \leftarrow \text{mutacao}(cl, \alpha);$ 
     $f_{ob_{cl}} \leftarrow \text{avalilar}(cl);$ 
    if  $f_{ob_{cl}} < f_{ob_{iteracao}^i}$  then
         $pop_{iteracao}^i \leftarrow cl;$ 
         $f_{ob_{iteracao}^i} \leftarrow f_{ob_{cl}};$ 
    end if
end for
end for
for  $i = \text{round}(B \cdot N)$  até  $N$  do
     $pop_{iteracao}^i \leftarrow \text{gerar anticorpo};$ 
     $f_{ob_{iteracao}^i} \leftarrow \text{avalia}(pop_{iteracao}^i);$ 
end for
if  $\text{resto}(\frac{iteracao}{atualiza-pop_{ext}}) = 0$  then
     $pop_{ext} \leftarrow \text{agrupa}(pop_{ext}, pop_{iteracao});$ 
     $pop_{ext} \leftarrow \text{supressao}(pop_{ext});$ 
end if
    Intervalo onde é aplicado o MBL2 para gerar o AM2 ou o AM3
if  $\text{resto}(\frac{iteracao}{k}) = 0$  then
     $pop_{iteracao} \leftarrow \text{MBL2}(pop_{iteracao});$ 
     $f_{ob_{iteracao}} \leftarrow \text{avalia funcao objetivo}(pop_{iteracao});$ 
end if
end while
 $pop \leftarrow \text{agrupa}(pop_{ext}, pop_{iteracao});$ 
 $pop \leftarrow \text{supressao}(pop_{ext});$ 
    Intervalo onde é aplicado o MBL1 para gerar o AM1 ou o AM3
 $pop \leftarrow \text{MBL1}(pop);$ 
 $f_{ob} \leftarrow \text{avalia funcao objetivo}(pop);$ 

```

Algoritmos mais bem adaptados podem ser obtidos fazendo-se uma avaliação de diferentes tipos de métodos de busca local, bem como o posicionamento desta busca e a periodicidade.

3.5 Análise de Sensibilidade Multiobjetivo das Soluções

A Análise de Sensibilidade Multiobjetivo considera cada uma das soluções obtidas pelos algoritmos de otimização (contando ótimo e sub-ótimos) como uma solução candidata à implementação. No caso do projeto de redes de distribuição de energia elétrica com incertezas na evolução das cargas, a análise é feita através de variações feitas nas condições de operação do sistema, que são as mudanças na configuração das cargas e no preço da energia elétrica. Estas variações nas condições de operação do sistema são obtidas via *Simulação de Monte Carlo* (Manno, 1999) (veja a Sec. 3.5.4 para mais informações sobre Simulações de Monte Carlo).

A importância desta análise se dá pelo fato de o cenário mais provável para o qual a rede foi projetada ser, geralmente, muito diferente do cenário real observado *a posteriori*. Com a análise de sensibilidade observa-se que algumas redes apresentam um desempenho muito ruim para cenários distintos ao cenário mais provável, indicando que estas redes são muito específicas e não apresentam robustez para lidar com variações na condição de operação do sistema. No entanto, outras redes mantêm desempenho satisfatório tanto no cenário médio quanto para cenários diferentes. Isto indica robustez adequada para garantir um bom desempenho para condições de operação diferentes.

A análise de sensibilidade multiobjetivo consiste, então, em gerar um conjunto de cenários diferentes do cenário mais provável através de Simulação de Monte Carlo. Para cada um destes cenários é feita a avaliação das soluções, dados os seguintes critérios:

1. Custo original da rede (f_1): custo da rede no cenário mais provável. É a avaliação da rede para o cenário utilizado durante a execução do algoritmo para a obtenção das soluções ótimas;
2. Taxa de infactibilidade da rede (f_2): por este critério é possível analisar a percentagem do número de cenários para os quais a rede foi infactível. O critério pode ser relacionado com a probabilidade de infactibilidade da rede para o horizonte de tempo considerado;

3. Custo médio da solução (f_3): este critério indica o custo médio da rede nos cenários para os quais ela é factível. Este critério difere-se do critério f_1 uma vez que ele representa um custo esperado para a rede quando a rede está sobre efeito de incertezas;
4. Custo médio das faltas (f_4): este critério analisa o custo inerente às falhas que podem ocorrer na rede para uma dada condição de carga. O cálculo deste critério é a média dos custos relativos à falhas nos cenários em que a rede é factível.

Estes critério podem ser expressos por:

$$f_1 = CN_{nm}^N \quad (3.4)$$

$$f_2 = \frac{1}{N_s} \sum_{i=1}^{N_s} FE_i^N \quad (3.5)$$

$$f_3 = \frac{1}{N_{fs}^N} \sum_{i=1}^{N_{fs}^N} CN_i^N \quad (3.6)$$

$$f_4 = \frac{1}{N_{fs}^N} \sum_{i=1}^{N_{fs}^N} FN_i^N \quad (3.7)$$

Onde:

CN_{nm}^N é o custo da rede N para as condições nominais de carga (cenário mais provável);

N_s é o número de cenários analisados;

FE_i^N é 1 se a rede N é infactível para a condição de carga do cenário i ou 0 caso contrário;

N_{fs}^N é o número de cenários onde a rede N é factível;

CN_i^N é o custo da rede N no cenário i ;

FN_i^N é o custo associado à faltas da rede N para o cenário i (este custo é calculado usando a equação (3.8)).

$$FN^N = \sum_{j=1}^m Y_i^N [\lambda(N_j).l(N_j).r(N_j).P(N_j).en_{tx}] \quad (3.8)$$

Onde:

Y_i^N é 1 se a rede N contém a conexão j ou 0 caso contrário;

$\lambda(N_j)$ é a taxa de falha da conexão j na rede N ;

$l(N_j)$ é o comprimento da conexão j na rede N ;

$r(N_j)$ é a duração média da falta da conexão j na rede N ;

$P(N_j)$ é a potência ativa que flui pela conexão j na rede N ;

en_{tx} é o preço da energia.

Como na análise de sensibilidade existem quatro critérios que devem ser atendidos, surge a necessidade de se fazer uma análise multiobjetivo de dominância entre as soluções. O conjunto formado pelas soluções não-dominadas é chamado de Conjunto Pareto-ótimo. Matematicamente, o conjunto Pareto-ótimo é definido como: considere a minimização da função vetorial, $f(\cdot)$, (o vetor de m funções objetivo) no qual o conjunto F representa o conjunto factível do problema (conjunto que contém as possíveis soluções para o problema). Não há, em geral, um único ponto $x \in F$ em que $f(\cdot)$ alcança o valor mínimo para todas as componentes. Então:

$$X^* = \{x^* \in F \mid \nexists z \in F \text{ tal que } f(x^*) \leq f(z) \text{ e } f(z) \neq f(x^*)\}; \quad (3.9)$$

no qual os operadores relacionais \leq e \neq são definidos, para os vetores $u, v \in \mathbb{R}^m$, como:

$$u \leq v \Leftrightarrow u_i \leq v_i \quad \forall i = 1, 2, \dots, m \quad (3.10)$$

$$u \neq v \Leftrightarrow u_i \neq v_i \quad \text{para alguns } i = 1, 2, \dots, m \quad (3.11)$$

Os pontos $x \in F$ que não pertencem ao conjunto X^* são ditos dominados, uma vez que há alguns outros pontos, $z \in F$, tais que $f(z) \leq f(x)$ e $f(z) \neq f(x)$, o que quer dizer que $f(z)$ é melhor que $f(x)$ em pelo menos uma coordenada, sem ser pior em

qualquer outra coordenada. Neste caso, z domina x . As soluções x^* que pertencem ao conjunto X^* são ditas soluções eficientes, uma vez que não são dominadas por nenhum outro ponto. Nos pontos eficientes todas as coordenadas (objetivos) não são piores em comparação com as coordenadas dos pontos não eficientes que eles dominam, mas não necessariamente em comparação com as coordenadas de todos os pontos não eficientes. A característica que determina que um ponto seja não eficiente é que ele é dominado por algum ponto eficiente, não por todos os pontos eficientes.

Para a utilização da Simulação de Monte Carlo são necessários os parâmetros das variáveis consideradas como incertezas no problema. Neste trabalho, serão utilizadas como variáveis incertas a carga futura em cada nó do sistema e o preço da energia elétrica em cada um destes nós. A seguir são apresentados os modelos dessas variáveis.

3.5.1 Modelo de Crescimento da Carga

A incerteza na evolução da carga é a principal fonte de incerteza presente no projeto de redes de distribuição de energia elétrica a longo prazo. Foi usada uma distribuição de probabilidade normal para modelar a incerteza na evolução da carga em cada nó.

3.5.2 Modelo de Variação do Preço da Energia

O preço da energia elétrica em cada nó do sistema para cada intervalo de tempo também é uma fonte de incerteza para o projeto a longo prazo de redes de distribuição de energia elétrica. A variação do preço da energia em cada nó se dá por fatores tais como a demanda em cada nó e os custos associados ao sistema de distribuição secundário. Para modelar a variação do preço da energia elétrica em cada nó do sistema foi utilizado uma distribuição de probabilidade normal.

3.5.3 Distribuição de Probabilidade Conjunta das Incertezas

Seja I o vetor das variáveis incertas do sistema, e I_i o vetor de variáveis incertas no ano i . O vetor I_{i+1} , das variáveis incertas para o próximo ano é calculado por (3.12):

$$I_{i+1} = I_i + \Delta_i \quad (3.12)$$

Onde:

Δ_i é um vetor de variáveis aleatórias que representa a variação de cada variável do problema, de um ano para o ano seguinte.

Para representar a dependência entre as variáveis, por exemplo a dependência entre o crescimento de carga entre duas regiões vizinhas, ou a dependência entre o crescimento da carga e o aumento do preço da energia, o vetor Δ_i é calculado por 3.13:

$$\Delta_i = D \cdot \delta_i \quad (3.13)$$

onde δ_i é um vetor de variáveis aleatórias independentes e D é a matriz que representa a relação entre as variáveis incertas. Na *Simulação de Monte Carlo*, o vetor δ_i é gerado segundo a distribuição de probabilidade adotada para cada variável incerta do sistema. δ_i é então utilizado para obter Δ_i e conseqüentemente I_{i+1} .

O efeito da incerteza acumulada sobre uma variável δ_i ao longo de 5 anos é ilustrado na figura 3.2.

3.5.4 Simulação de Monte Carlo

A Simulação de Monte Carlo é uma técnica computacional que gera aleatoriamente uma série de valores para as variáveis incertas de um modelo. Esses valores gerados

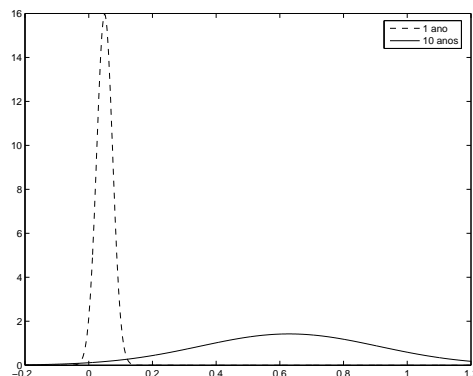


Figura 3.2: Na figura é mostrado um modelo de crescimento de carga anual com $\bar{x}_1 = 0.05$ e $\sigma_1 = 0.025$ e seu modelo equivalente para $t_a = 10$ anos, com $\bar{x}_1 = 0.629$ e $\sigma_1 = 0.28$. O aumento na incerteza do modelo é perceptível.

aleatoriamente seguem uma distribuição de probabilidade previamente estabelecida (Papoulis, 1991).

Em cada simulação são analisados múltiplos cenários possíveis, que seguem as distribuições de probabilidade dadas para cada variável incerta. Essa avaliação de múltiplos cenários tem como objetivo analisar algum aspecto do desempenho do sistema tais como estabilidade, confiabilidade, robustez, etc..

Para o projeto de redes, a Simulação de Monte Carlo utiliza as distribuições conjuntas de probabilidade de cada nó do sistema para criar um conjunto de cenários de configuração de carga e assim avaliar o desempenho de cada rede candidata em conformidade com a Análise de Sensibilidade Multiobjetivo descrita na seção 3.5.

Capítulo 4

Resultados

Os resultados apresentados neste capítulo se referem às simulações feitas para o projeto de redes de distribuição de energia elétrica com incerteza na evolução da carga para horizonte a longo prazo. Os métodos de otimização empregados para a obtenção das soluções ótimas foram o Algoritmo Clonal e os algoritmos meméticos: AM1, AM2 e AM3. Para tratar as incertezas na evolução da carga foi utilizada uma análise de sensibilidade multiobjetivo.

Associada à cada nó do sistema existe uma distribuição conjunta de probabilidade, representando o crescimento da carga e as variações no preço da energia elétrica. Adotou-se para todos os nós do sistema a distribuição normal tanto para o crescimento da carga quanto para a variação do preço da energia. Os dados destas distribuições estão apresentados na tabela 4.1. Existem 9 possíveis tipos de condutores para o projeto da rede. Os dados dos condutores estão apresentados na tabela 4.2. Para a codificação

Taxa de crescimento da carga	1 ano	10 anos
média:	0,050	0,629
desvio padrão:	0,025	0,280
Variação do preço da energia	1 ano	10 anos
média:	0,000	0,000
desvio padrão:	0,050	0,629

Tabela 4.1: Parâmetros das distribuições de probabilidades

Tabela 4.2: Dados dos condutores utilizados no projeto de redes de distribuição

Tipo de condutor	resistência (Ω/km)	reatância (Ω/km)	potência nominal (kW)
1	0,4853	1,6118	2817
2	0,4679	1,0145	3776
3	0,4505	0,6375	5115
4	0,4428	0,5205	5975
5	0,4331	0,4019	6860
6	0,4237	0,3184	7983
7	0,4026	0,2006	1082
8	0,3037	0,8220	4278
9	0,2567	0,2646	8652

das variáveis foi utilizado o algoritmo *Controlled-Greedy Encoding* (Sec. 1.3.1).

Primeiramente são apresentados na seção 4.1 os resultados referentes à simulação para um sistema teste de 21 nós com a utilização dos seguintes algoritmos: Algoritmo Clonal (2.1); *AM1*; *AM2*; *AM3* e *AM4* (3.4) e uma análise de desempenho destes algoritmos. Depois será apresentado na Sec. 4.2 o resultado da simulação do projeto de redes de distribuição de energia elétrica com incerteza na evolução da carga para um sistema teste de 100 nós. Neste caso, será feita a simulação apenas com o *AM3* e com o algoritmo clonal. A intenção é demonstrar a validade do Algoritmo Memético para outros problemas além do caso de 21 nós. Os outros algoritmos meméticos não foram utilizados uma vez que o *AM3* apresentou os melhores resultados para o sistema de 21 nós.

Para a previsão do cenário mais provável, aquele que é empregado no processo evolutivo do algoritmo de otimização, utiliza-se a média de crescimento anual da carga de cada nó do sistema e de variação do preço da energia elétrica, dados na tabela 4.1. Sobre este cenário aplica-se o algoritmo de otimização que terá como resposta um conjunto de soluções para o problema, dado o horizonte de tempo considerado.

As soluções obtidas pelo algoritmo de otimização para o cenário mais provável são então levadas à análise de sensibilidade multiobjetivo (Sec. 3.5) para sua avaliação quanto ao desempenho em cenários diferentes. Assim, geram-se cenários em que a carga e o custo da energia elétrica em cada nó são diferentes dos outros cenários estudados anteriormente. Para a análise de sensibilidade empregada, são obtidos 2500 cenários

distintos, sobre os quais são avaliados os desempenhos de todas as redes pertencentes ao conjunto solução do algoritmo de otimização. Os critérios a serem atendidos na análise de sensibilidade foram apresentados na Sec. 3.5.

Uma vez obtido o Conjunto Pareto-ótimo, basta ao projetista, adotando um critério de escolha conveniente ao projeto, escolher entre todas as redes qual melhor se aplica aos requisitos do projeto. Neste trabalho, é aplicada uma decisão que *a priori* estabelece como redes aplicáveis aquelas redes que apresentam uma taxa de infactibilidade menor ou igual a 20%.

Vale ressaltar que o número reduzido de vezes em que cada caso foi executado se deve ao alto custo computacional associado à execução do algoritmo de otimização. Para a execução completa do caso em que foi empregado o *AM3*, foram gastas aproximadamente 43 horas de trabalho de um processador Intel Core 2 Duo T5500; 1,0 GB de memória RAM utilizando o software *Matlab 7.0*¹. Por este motivo não foi possível fazer um tratamento estatístico do desempenho dos Algoritmos Meméticos utilizados para o projeto de redes de distribuição de energia elétrica com incertezas na evolução da carga para horizonte a longo prazo.

4.1 O Sistema de 21 nós

Um sistema de 21 nós foi proposto em Carrano et al. (2007a) para o projeto de redes de distribuição de energia elétrica com incertezas na evolução da carga. Cinco destes nós pertencem a uma rede já existente, mostrada na figura 4.1. Por ser uma rede que já existe, os custos relativos às mudanças ocorridas serão custos relacionados à substituição das conexões existentes e a reconexão dos nós. Os outros 16 nós deverão ser conectados ao sistema de forma que o projeto final seja uma rede conexa. Com isso, exige-se do projeto de redes não apenas a inclusão de novos nós ou a construção de toda a rede. É necessário também a readequação da rede já existente às novas condições de operação do sistema.

¹*Matlab* é marca registrada da *Mathworks*.

Utilizando a técnica *Controlled-Greedy Encoding* de redução do espaço de redes, com $mxv = 6$ e $mnv = 4$, obtém-se um grafo de busca com 62 variáveis, como mostrado na figura 4.2. O horizonte de tempo considerado é de 10 anos e os dados das distribuições de probabilidade para o crescimento da carga e para a variação do preço de energia são apresentados na tabela 4.1.



Figura 4.1: Sistema de 21 nós - configuração inicial

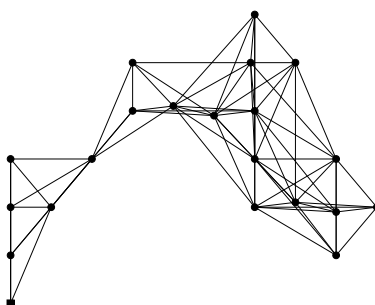


Figura 4.2: Sistema de 21 nós - codificação pela *Controlled-Greedy Encoding*

4.1.1 Resultados Obtidos via Algoritmo Clonal

Os resultados apresentados nesta seção foram obtidos por meio de duas execuções do Algoritmo Clonal, usando o cenário mais provável. Em seguida é feita a análise de sensibilidade multiobjetivo. Os parâmetros do algoritmo utilizado para as simulações estão apresentados na tabela 4.3.

Tabela 4.3: Parâmetros utilizados para a implementação do Algoritmo Clonal

Parâmetro	Valor
Critério de parada	executar 200 gerações
Número de indivíduos na população	50
B	60%
r_b	20
β	2

As soluções obtidas pelas duas simulações realizadas que apresentam taxa de infactibilidade menor ou igual à 20% estão apresentadas na tabela 4.4. Para as duas simulações foi encontrada a mesma rede como sendo a solução ótima para o cenário mais provável, solução $1cl$, apresentada na figura 4.3.

Tabela 4.4: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo com taxa de infactibilidade não superior a 20% - Simulação com o Algoritmo Clonal

Solução	f_1	f_2	f_3	f_4
Ótimo para o cenário mais provável				
$1cl$	1105002,59	0,7190	1213229,95	1732476,16
Primeira simulação				
$50cl^A$	1192030,00	0,0060	1309338,10	1721028,42
$66cl^A$	1201313,78	0,0060	1318427,57	1682633,80
$151cl^A$	1232045,53	0,0060	1349438,69	1667692,85
$225cl^A$	1277689,56	0,0498	1396528,59	1666312,46
$228cl^A$	1282720,65	0,0045	1396296,91	1651553,65
$264cl^A$	1358756,61	0,0120	1493389,08	1309433,98
Segunda simulação				
$23cl^B$	1165360,41	0,0032	1284710,10	1755828,40
$217cl^B$	1234800,30	0,0348	1349825,97	1643260,24
$238cl^B$	1242314,19	0,0032	1357270,99	1690033,59
$278cl^B$	1301154,97	0,0524	1417501,15	1634521,30
$312cl^B$	1390792,81	0,0100	1515806,79	1544348,15
$405cl^B$	1594019,08	0,0032	1714264,60	1630519,64
$406cl^B$	1594767,67	0,0032	1714967,94	1630506,23
$411cl^B$	1609558,32	0,0012	1731990,43	2023950,20
$419cl^B$	1644176,91	0,0200	1781503,90	1368106,78

Na primeira simulação, foram encontradas 436 soluções ótimas, com um custo computacional de 80.632 avaliações da função objetivo. Estas soluções estão representadas na figura 4.4, onde as soluções foram ordenadas por ordem crescente da função objetivo. Após serem levadas à análise de sensibilidade multiobjetivo, foram encontradas 39 soluções não-dominadas (tabela A.1 do apêndice A). Isto indica que o restante das soluções são dominadas por estas e não apresentam escolhas razoáveis de imple-

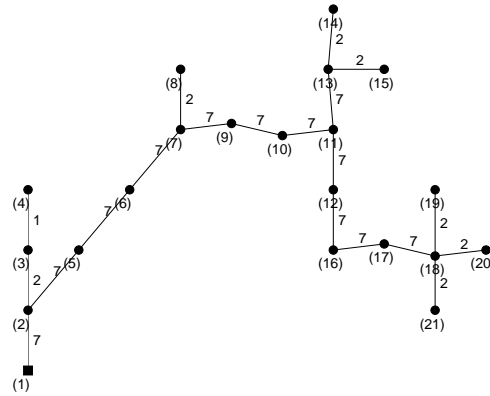


Figura 4.3: Rede ótima para o cenário mais provável

mentação. Das 39 soluções não-dominadas, apenas seis ($50cl^A$, $66cl^A$, $151cl^A$, $225cl^A$, $228cl^A$ e $264cl^A$) apresentam taxa de ineficiência menor ou igual à 20%, mostradas na tabela 4.4. Embora seja o ótimo encontrado para o cenário mais provável, a rede $1cl$, apresenta um resultado muito ruim quando é analisado seu desempenho para os cenários distintos criados pela simulação de Monte Carlo. O critério f_2 indica que esta rede apresenta uma taxa de ineficiência próxima a 70%. Ela não é robusta o suficiente para suportar uma variação nas condições de operação do sistema.

Na segunda simulação foi obtido um conjunto com 442 soluções ótimas, apresentadas na figura 4.7, com um total de 80.300 avaliações de função objetivo. A solução ótima encontrada para o cenário mais provável é a mesma encontrada na primeira simulação, solução $1cl$ da tabela 4.4. Após levar o conjunto de soluções à análise de sensibilidade multiobjetivo, foram encontradas 46 soluções não-dominadas, apresentadas na tabela A.1 do apêndice A. Destas 46 soluções, nove apresentam taxa de ineficiência menor ou igual à 20% e estão apresentadas na tabela 4.4. São as seguintes soluções: $23cl^B$, $217cl^B$, $238cl^B$, $278cl^B$, $312cl^B$, $405cl^B$, $406cl^B$, $411cl^B$ e $419cl^B$.

Se a prioridade do projeto é uma rede que seja a mais robusta frente às variações das condições de operação do sistema, o que é representado na análise de sensibilidade pelo critério f_2 (taxa de ineficiência), a rede a ser escolhida para a primeira simulação

é a rede $50cl^A$, figura 4.5. Esta rede ainda tem um custo muito próximo da rede do cenário mais provável. Ela tem um custo aproximadamente 7,8% superior à rede ótima para o cenário mais provável com uma taxa de inactibilidade de 0,60 %. Para a segunda simulação a rede escolhida é a rede $23cl^B$ (figura 4.8) que apresenta taxa de inactibilidade de 0,32% e custo para o cenário mais provável 5,46% mais caro que o custo da rede ótima.

Caso se deseje uma rede que apresente a melhor confiabilidade, isto é, que tenha os menores índices de falhas, a escolha é a rede $264cl^A$, figura 4.6. Ela apresenta o menor custo médio de falha para os cenários em que ela foi factível. Na segunda simulação a rede $419cl^B$, apresentada na figura 4.9, é a que apresenta menor custo médio com falhas.

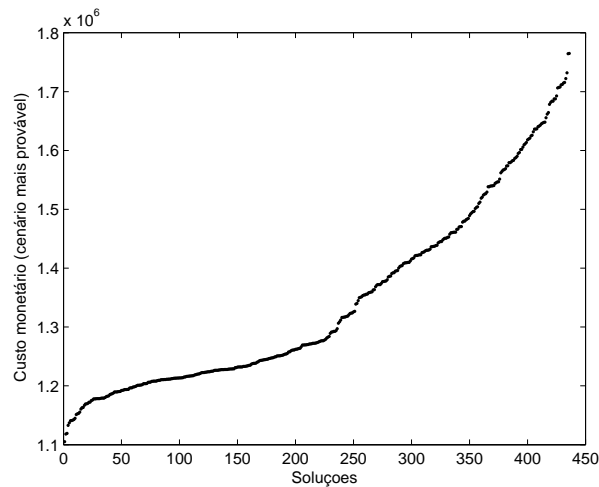


Figura 4.4: Redes obtidas pela otimização via Algoritmo Clonal - Primeira Simulação.

4.1.2 Resultados Obtidos via AM1

As soluções ótimas e sub-ótimas apresentadas nesta seção são obtidas pela simulação do AM1, descrito na seção 3.4. O critério de parada para o MBL1 é gerar consecutivamente 50 redes vizinhas sem que haja melhoria na solução (um critério de parada para indicar que foi obtido um ótimo local). Este algoritmo faz com que algumas soluções

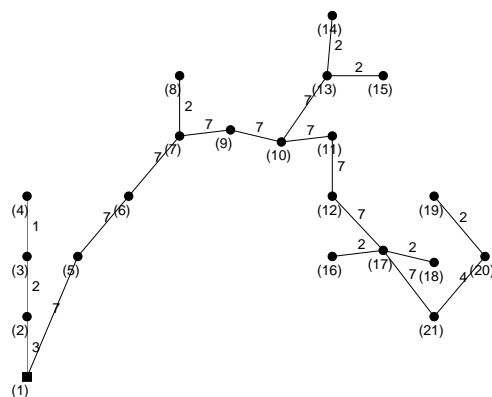


Figura 4.5: Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio) - Primeira Simulação com o Algoritmo Clonal

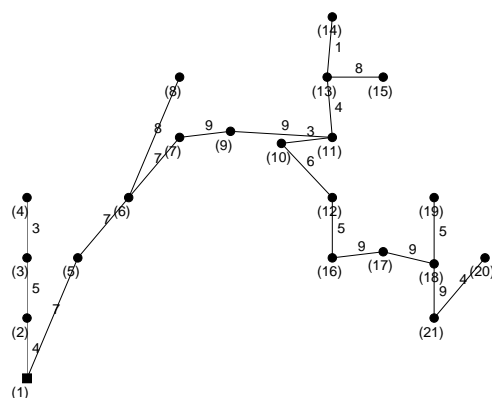


Figura 4.6: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Primeira Simulação com o Algoritmo Clonal

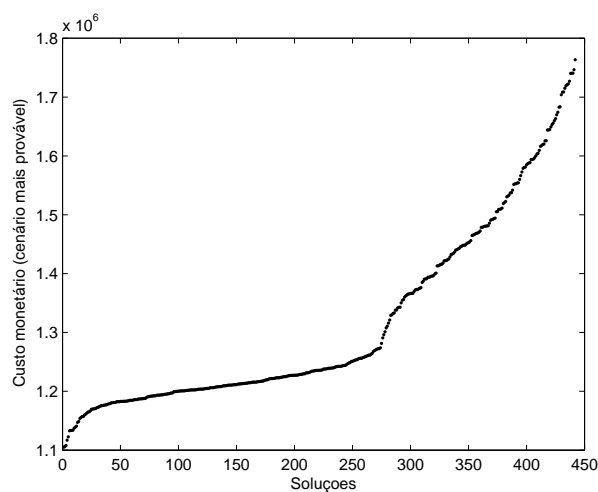


Figura 4.7: Redes obtidas pela otimização via Algoritmo Clonal - Segunda Simulação.

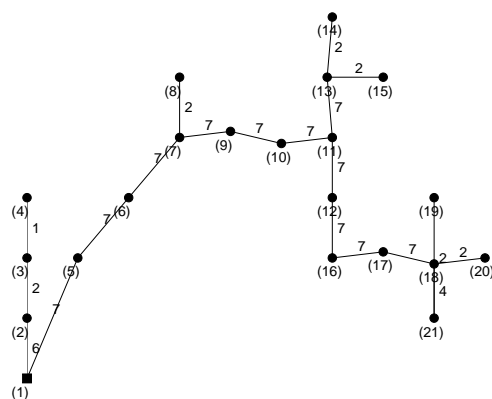


Figura 4.8: Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio) - Segunda Simulação com o Algoritmo Clonal

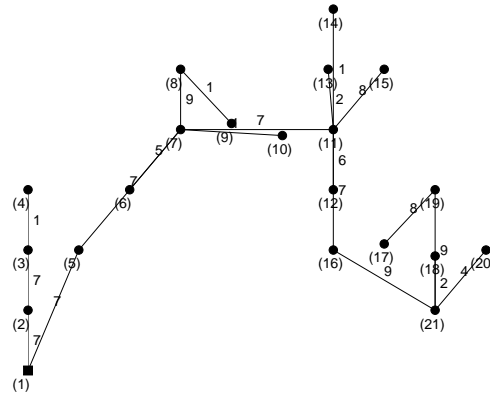


Figura 4.9: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade) - Segunda Simulação com o Algoritmo Clonal

após serem refinadas converjam para uma mesma solução em comum, ou seja, caminhem para um mesmo mínimo local. Ele aumenta as opções e melhora o desempenho das redes que podem ser utilizadas no projeto.

Nas duas simulações realizadas com o *AM1* foi obtida a mesma rede como sendo o ótimo para o cenário mais provável (rede *1am1* da tabela 4.5). Ela é a mesma rede encontrada com as simulações com o algoritmo clonal, apresentada na figura 4.3. Na primeira simulação foram encontradas 435 soluções, mostradas na figura 4.10. Elas estão ordenadas em ordem crescente do valor da função objetivo. Para obter estas soluções foram necessárias 213.155 avaliações de função objetivo, sendo 81.210 gastas durante a execução do algoritmo clonal e 131.945 gastas pela execução do MBL1. Estas soluções após levadas à análise de sensibilidade, retornam um conjunto Pareto-ótimo com 49 soluções (tabela A.3 do apêndice A), das quais oito (*57am1^A*, *82am1^A*, *95am1^A*, *237am1^A*, *238am1^A*, *315am1^A*, *383am1^A* e *430am1^A*) têm taxa de infactibilidade não superior a 20%, apresentadas na tabela 4.5.

Com a segunda simulação foram obtidas 442 soluções para o cenário mais provável, mostradas na figura 4.13. Para encontrar estas soluções foram necessárias 203.970 avaliações de função objetivo, sendo 80.847 gastas durante a execução do algoritmo clonal e 123.123 gastas pela execução do MBL1. Estas soluções após levadas à análise de sensi-

bilidade retornam um conjunto Pareto-ótimo com 41 soluções (tabela A.4 do apêndice A), das quais seis ($19am1^B$, $218am1^B$, $236am1^B$, $251am1^B$, $313am1^B$ e $375am1^B$) têm taxa de infactibilidade menor ou igual à 20%, apresentadas na tabela 4.5.

O projetista se quiser uma rede com uma boa robustez às variações do sistema e ainda com um custo para o cenário mais provável próximo do custo da rede ótima, deve escolher, na primeira simulação, a rede $57am1^A$, figura 4.11. O custo desta rede para o cenário médio é 1,98% maior que o custo da rede ótima e ela apresenta taxa de infactibilidade de 0,39%. Para a segunda simulação, a melhor opção é a rede $19am1^B$, figura 4.14, pois ela apresenta um custo apenas 1,29% maior do que aquela do cenário mais provável e uma taxa de infactibilidade de apenas 0,28%.

Caso o projetista tenha prioridade por uma rede que apresente a maior confiabilidade, ou seja, apresente o menor custo médio por falhas, a melhor alternativa para a primeira simulação é a rede $430am1^A$, figura 4.12. Esta rede tem o menor custo médio por falhas dentre as redes disponíveis. Para a segunda simulação a escolha seria a rede $375am1^B$, figura 4.15.

Tabela 4.5: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM1

Solução	f_1	f_2	f_3	f_4
Ótimo para o cenário mais provável				
$1am1$	1105002,59	0,6987	1211782,54	1726414,17
Primeira Simulação				
$57am1^A$	1126893,34	0,0038	1246994,74	1736906,28
$82am1^A$	1133810,35	0,0038	1255656,73	1615977,87
$95am1^A$	1136752,37	0,0038	1259746,55	1602717,09
$237am1^A$	1160659,78	0,0038	1281530,18	1402110,99
$238am1^A$	1160711,33	0,1995	1278617,37	1450056,06
$315am1^A$	1176779,82	0,1946	1293143,01	1156908,62
$383am1^A$	1199906,34	0,0121	1321506,58	1372324,45
$430am1^A$	1266126,28	0,0024	1378026,51	1697831,21
Segunda Simulação				
$19am1^B$	1119287,86	0,0028	1238269,06	1689470,33
$218am1^B$	1159218,27	0,0028	1280493,77	1646169,48
$236am1^B$	1162656,87	0,1883	1281516,62	1425793,14
$251am1^B$	1165036,26	0,0028	1285005,87	1307260,78
$313am1^B$	1175662,68	0,0024	1289143,81	1669485,96
$375am1^B$	1192514,69	0,1923	1313375,67	1272270,20

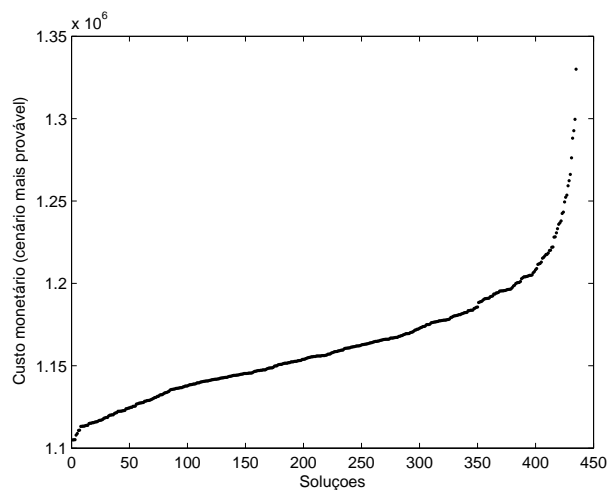


Figura 4.10: Redes obtidas pela otimização via AM1 - Primeira Simulação

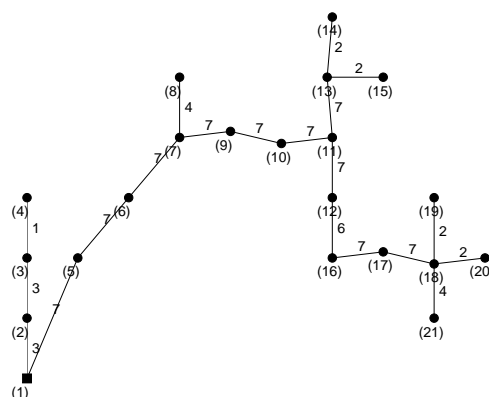


Figura 4.11: Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Primeira Simulação com o AM1

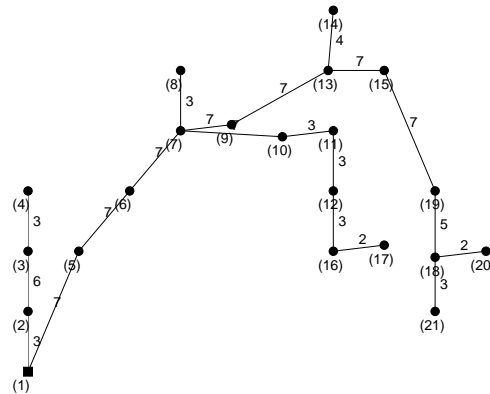


Figura 4.12: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)-Primeira Simulação com o AM1

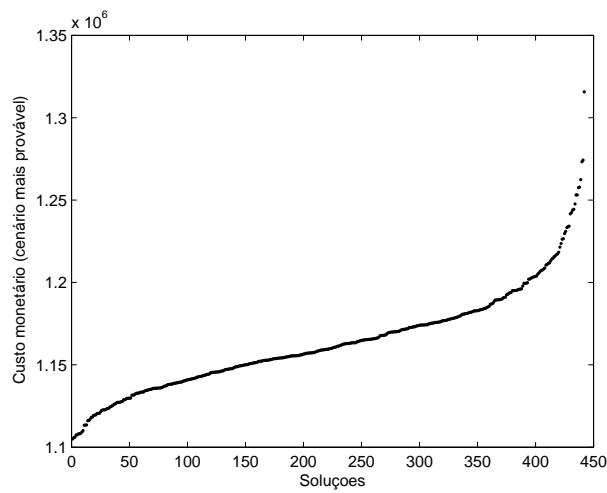


Figura 4.13: Redes obtidas pela otimização via AM1 - Segunda Simulação

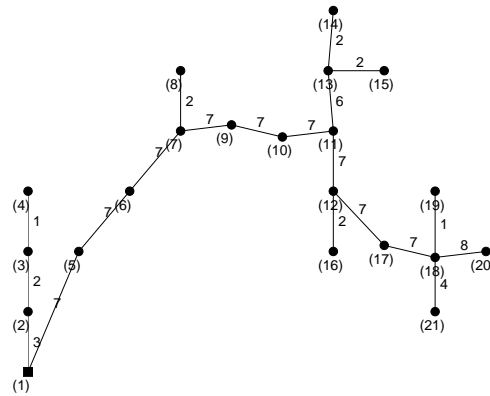


Figura 4.14: Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Segunda Simulação com o AM1

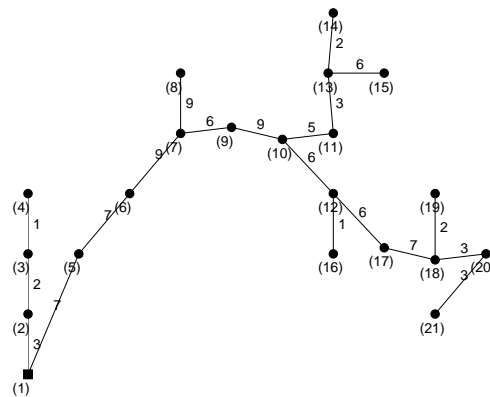


Figura 4.15: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Segunda Simulação com o AM1

4.1.3 Resultados Obtidos via AM2

Os resultados apresentados a seguir foram obtidos com a utilização do algoritmo memético *AM2*, descrito na seção 3.4. O *MBL2* é utilizado aqui em intervalos de 10 em 10 gerações do algoritmo clonal, sendo aplicado a todas as soluções da população de anticorpos. A escolha do intervalo de aplicação do *MBL2* foi feita para 10 gerações após serem realizados alguns testes com outros intervalos: intervalos maiores que 10 gerações não apresentaram bons resultados e intervalos menores não apresentaram melhorias significativas nos resultados.

Com a primeira simulação do algoritmo *AM2* obteve-se 508 soluções (figura 4.16). Para isso foram necessárias 235.802 avaliações de função objetivo sendo 159.528 utilizadas pelo algoritmo clonal e 76.274 pelo método de busca local *MBL2*. O número elevado de avaliações utilizadas pelo AC, em comparação aos outros algoritmos, se dá pelo fato de que o *AM2* é capaz de encontrar o número máximo de anticorpos (soluções) em meados do número máximo de gerações. Todas estas soluções foram levadas à análise de sensibilidade multiobjetivo, retornando um total de 25 soluções pertencentes ao conjunto Pareto-ótimo, mostradas na tabela A.5 do apêndice A. Das 25 soluções não-dominadas, 12 têm taxa de infactibilidade menor ou igual à 20%. São as soluções : $4am2^A$, $9am2^A$, $129am2^A$, $132am2^A$, $242am2^A$, $243am2^A$, $287am2^A$, $320am2^A$, $374am2^A$, $396am2^A$, $412am2^A$, e $427am2^A$. Elas estão apresentadas na tabela 4.6.

Para a segunda simulação foram necessárias 225.078 avaliações de função objetivo sendo 72.330 utilizadas pelo algoritmo clonal e 152.748 pelo método de busca local *MBL2*, para obter as 501 soluções apresentadas na figura 4.19. Destas 501 soluções, 31 são soluções não dominadas (tabela A.6 do apêndice A), sendo que 7 destas têm taxa de infactibilidade menor ou igual à 20%. São as soluções: $2am2^B$, $4am2^B$, $5am2^B$, $10am2^B$, $35am2^B$, $62am2^B$, e $498am2^B$ (tabela 4.6). Para as duas simulações foi encontrada a mesma rede ótima para o cenário mais provável (solução $1am2$ da tabela 4.6), que é a mesma solução encontrada nas demais simulações. A rede ótima obtida nas duas

simulações é apresentada na figura 4.3.

Para o projetista priorizar ao mesmo tempo o custo para o cenário mais provável e a taxa de infactibilidade, ou seja, obter uma rede que dê um baixo custo para o cenário mais provável e também tenha uma baixa taxa de infactibilidade, ele tem como melhor escolha, para a primeira simulação, a solução $4am2^A$ (figura 4.17). Esta apresenta um custo para o cenário mais provável de 0,66% mais caro que o custo da rede ótima e tem uma taxa de infactibilidade de apenas 0,30%. Para a segunda simulação, o projetista deve optar pela solução $2am2^B$, que apresenta um custo para o cenário mais provável de 0,19% mais caro que o custo da rede ótima para o novo cenário e tem uma taxa de infactibilidade de aproximadamente 0,31%.

Já quando o projetista prioriza a confiabilidade da rede escolhida, o que é indicado pelo critério f_4 (custo médio para falhas), ele deve escolher a rede $427am2^A$ (figura 4.18) da tabela 4.6 para a primeira simulação. Na segunda simulação o projetista deve escolher a rede $498am2^B$ da tabela 4.6, apresentada na figura 4.21.

Tabela 4.6: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM2 - Primeira Simulação

Solução	f_1	f_2	f_3	f_4
Ótimo para o cenário mais provável				
$1am2$	1105002,59	0,6834	1211575,99	1725468,46
Primeira Simulação				
$4am2^A$	1112306,93	0,0030	1227355,08	1681936,29
$9am2^A$	1124497,94	0,0030	1243666,16	1650669,73
$129am2^A$	1161912,09	0,1691	1277462,24	1239120,88
$132am2^A$	1148686,67	0,0030	1273377,85	1479660,94
$242am2^A$	1197230,52	0,1736	1313319,64	1123244,42
$243am2^A$	1185373,67	0,1804	1303592,70	1119295,34
$287am2^A$	1180626,96	0,1864	1301321,41	1147916,84
$320am2^A$	1219697,97	0,1518	1330957,37	1380485,03
$374am2^A$	1242324,99	0,0030	1365927,27	1327096,57
$396am2^A$	1284599,55	0,1503	1394309,08	1294269,42
$412am2^A$	1227816,33	0,0398	1350952,03	1452512,50
$427am2^A$	1323312,90	0,0030	1446609,21	1264496,49
Segunda Simulação				
$2am2^B$	1107128,93	0,0030	1226976,74	1732700,60
$4am2^B$	1110149,12	0,0030	1229935,45	1688689,55
$5am2^B$	1114596,86	0,0030	1231258,66	1626432,71
$10am2^B$	1121035,50	0,0030	1239987,28	1617068,47
$35am2^B$	1145090,61	0,0030	1269208,86	1412587,71
$62am2^B$	1154324,63	0,0061	1280126,19	1310008,19
$498am2^B$	1452151,61	0,0000	1581686,49	1337900,41

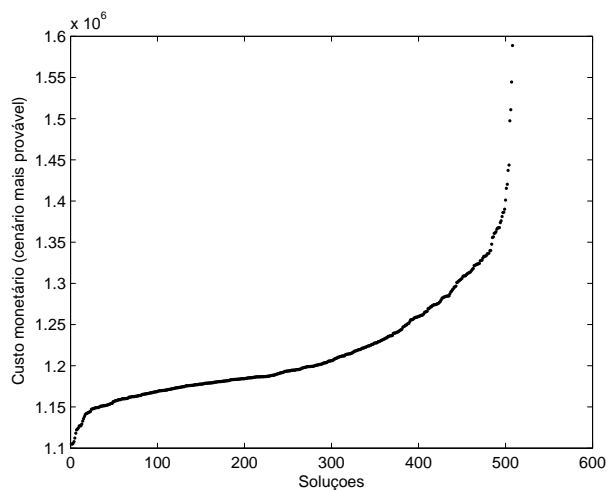


Figura 4.16: Redes obtidas pela otimização via AM2 - Primeira Simulação

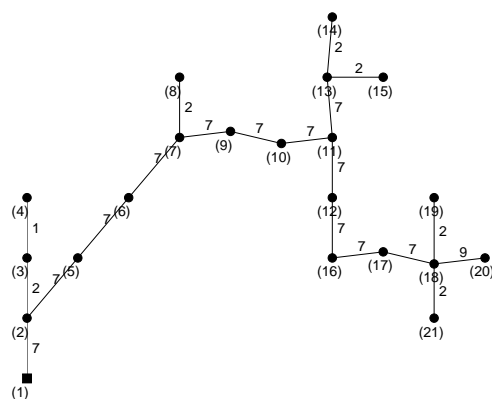


Figura 4.17: Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Primeira Simulação com o AM2

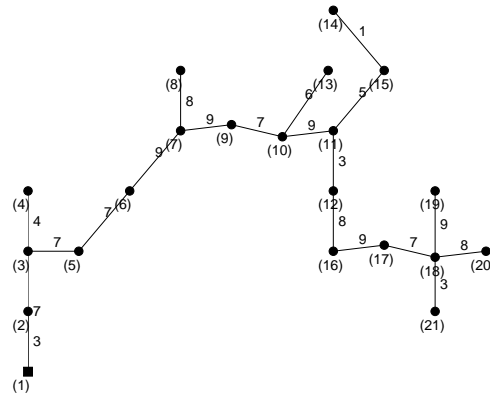


Figura 4.18: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)-Primeira Simulação com o AM2

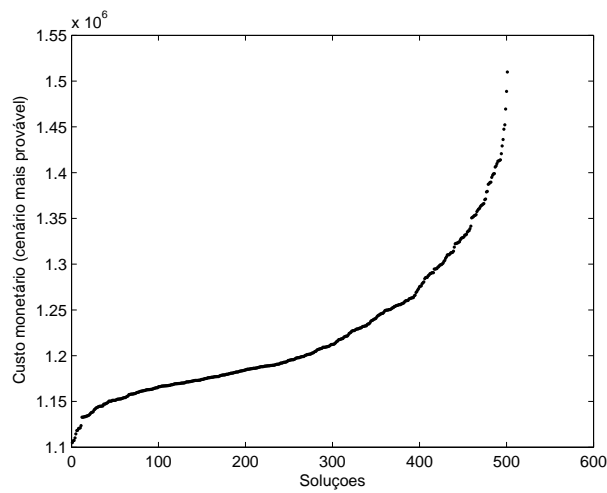


Figura 4.19: Redes obtidas pela otimização via AM2 - Segunda Simulação

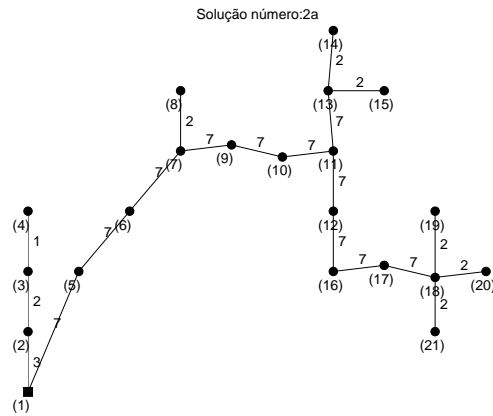


Figura 4.20: Rede escolhida para ser aplicada ao projeto (priorizando custo para cenário médio)- Segunda Simulação com o AM2

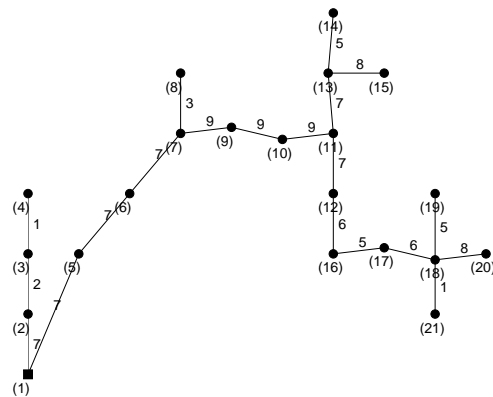


Figura 4.21: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)- Segunda Simulação com o AM2

4.1.4 Resultados Obtidos via AM3

Para obtenção destes resultados foi utilizado o algoritmo memético *AM3* (seção 3.4).

Nas duas simulações realizadas, foi encontrada como rede ótima para o cenário mais provável a rede *1am3* da tabela 4.7. Esta é a mesma rede encontrada em todas as outras simulações feitas para este sistema de 21 nós, apresentada na figura 4.3. Na primeira simulação foram necessárias 341.110 avaliações de função objetivo (sendo 159.528 avaliações realizadas pelo algoritmo clonal, 76.274 avaliações realizadas pelo MBL2 e 105.308 avaliações realizadas pelo MBL1) para obter 440 soluções. Estas soluções estão apresentadas na figura 4.22. Após ser feita a análise de sensibilidade multiobjetivo, obtém-se 35 soluções não-dominadas (tabela A.7 do apêndice A). Na tabela 4.7 estão apresentadas as 10 soluções não-dominadas obtidas que apresentam taxa de infactibilidade menor ou igual à 20% (*6am3^A*, *10am3^A*, *55am3^A*, *157am3^A*, *258am3^A*, *272am3^A*, *330am3^A*, *366am3^A*, *372am3^A* e *416am3^A*).

Para a segunda simulação, foram feitas 323.152 avaliações de função objetivo (153.313 avaliações realizadas pelo algoritmo clonal, 71.008 avaliações realizadas pelo MBL2 e 98.831 avaliações realizadas pelo MBL1) para obter um conjunto com 503 soluções (figura 4.25). Destas 503 soluções, 41 soluções são não-dominadas e estão apresentadas na tabela A.8 do apêndice A. Na tabela 4.7 estão apresentadas as 10 soluções não-dominadas que têm taxa de infactibilidade não superior à 20% (*2am3^B*, *14am3^B*, *15am3^B*, *29am3^B*, *51am3^B*, *103am3^B*, *260am3^B*, *301am3^B*, *356am3^B*, e *370am3^B*).

A rede *1am3* da tabela 4.7 não é uma boa alternativa de implementação pois tem taxa de infactibilidade muito alta, próxima à 70%. Num projeto em que a prioridade é manter o menor custo de instalação da rede conservando uma baixa taxa de infactibilidade, o projetista tem como uma boa opção, para a primeira simulação, a rede *6am3^A* da figura 4.23. Esta opção apresenta um custo de apenas 0,19% mais caro que o custo da rede ótima para o cenário mais provável com taxa de infactibilidade de 0,00%. Já para a segunda simulação, a melhor opção é a rede *2am3^B*, que é a mesma rede *6am3^A*

encontrada na primeira simulação com o AM3.

Caso o projeto requeira uma rede para qual a prioridade seja a confiabilidade, ou seja, apresente o menor custo médio por falhas, a melhor opção para a primeira simulação é a rede $416am3^A$ da figura 4.24. Ela apresenta o menor custo médio para falhas. Para a segunda simulação a melhor escolha é a rede $370am3^B$, apresentada na figura 4.26.

Tabela 4.7: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM3 - Primeira Simulação

Solução	f_1	f_2	f_3	f_4
Ótimo para o cenário mais provável				
$1am3$	1105002,59	0,6870	1212457,09	1728357,39
Primeira Simulação				
$6am3^A$	1107128,93	0,0000	1218338,25	1690987,33
$10am3^A$	1112148,35	0,0041	1226916,09	1679831,60
$55am3^A$	1128933,35	0,1835	1246263,44	1361452,22
$157am3^A$	1143376,76	0,0041	1262585,52	1537115,63
$258am3^A$	1155061,72	0,1822	1270195,86	1474318,20
$272am3^A$	1156281,49	0,1863	1274559,37	1227706,85
$330am3^A$	1164934,62	0,0041	1285145,61	1524281,15
$366am3^A$	1171117,81	0,0041	1288173,90	1491441,84
$372am3^A$	1171643,55	0,0041	1292614,17	1366650,29
$416am3^A$	1179673,76	0,1905	1299788,16	1178235,63
Segunda Simulação				
$2am3^B$	1107128,93	0,0000	1218338,25	1690987,33
$14am3^B$	1112255,29	0,0056	1228124,88	1681960,26
$15am3^B$	1112415,26	0,0056	1228186,18	1681873,95
$29am3^B$	1120419,49	0,0056	1239097,77	1629585,68
$51am3^B$	1125718,53	0,0056	1246283,87	1576847,77
$103am3^B$	1137096,43	0,0056	1255255,68	1528115,34
$260am3^B$	1159509,93	0,1819	1273662,50	1298224,61
$301am3^B$	1162040,45	0,0056	1277881,62	1522326,59
$356am3^B$	1169855,94	0,0056	1288084,76	1502394,07
$370am3^B$	1171879,51	0,0056	1294773,39	1257370,06

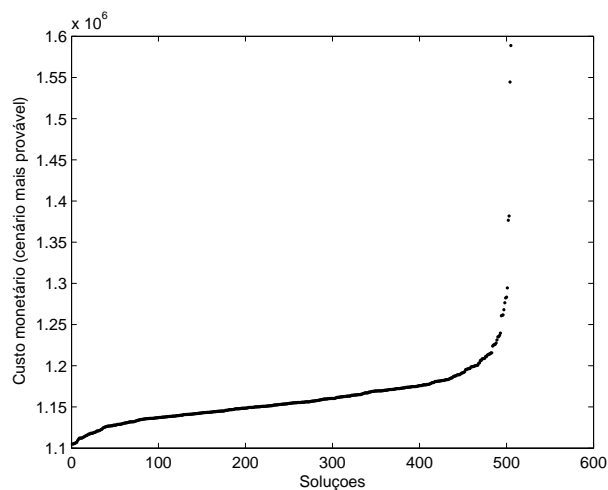


Figura 4.22: Redes obtidas pela otimização via AM3 - Primeira Simulação

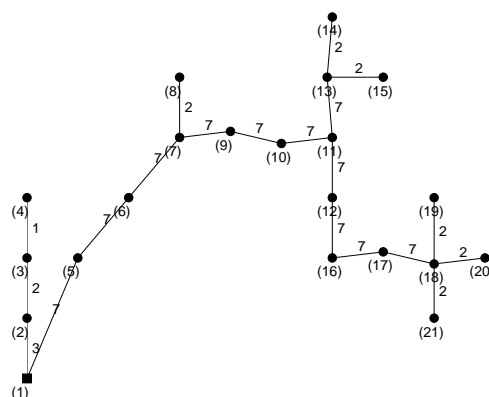


Figura 4.23: Rede escolhida para ser aplicada ao projeto (priorizando custo para o cenário médio)- Primeira e Segunda Simulação com o AM3

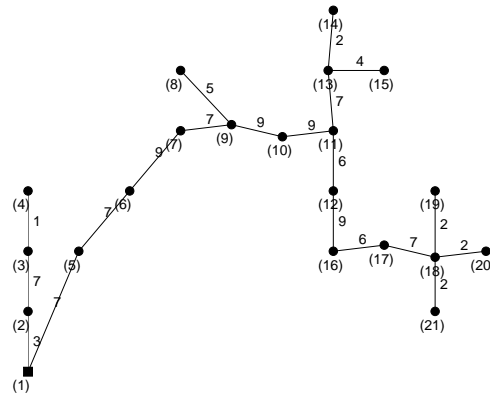


Figura 4.24: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)-Primeira Simulação com o AM3

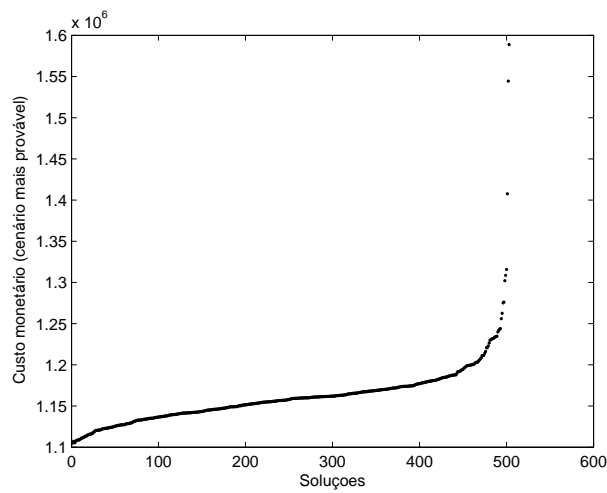


Figura 4.25: Redes obtidas pela otimização via AM3 - Segunda Simulação

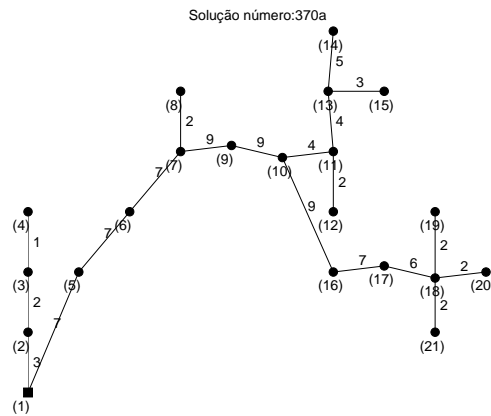


Figura 4.26: Rede escolhida para ser aplicada ao projeto (priorizando confiabilidade)-Segunda Simulação com o AM3

4.1.5 Comparação dos Resultados Obtidos para o Caso de 21 nós

A figura 4.27 apresenta as soluções obtidas pelas execuções realizadas para cada algoritmo. É fácil observar que o algoritmo clonal obteve um conjunto de soluções significativamente diferente para cada execução, enquanto os conjuntos de soluções obtidos pelos algoritmos meméticos apresentam-se bem próximos uns dos outros. O AM3 foi o que apresentou a menor variação nos conjuntos de soluções encontrados. Analisando pela média das soluções encontradas, conforme a tabela 4.8, observa-se que o AM3 é o que apresenta a menor variação da média para as execuções realizadas. Os outros algoritmos meméticos também são robustos, mas com pior média. O algoritmo clonal apresentou a maior variação e também a pior média.

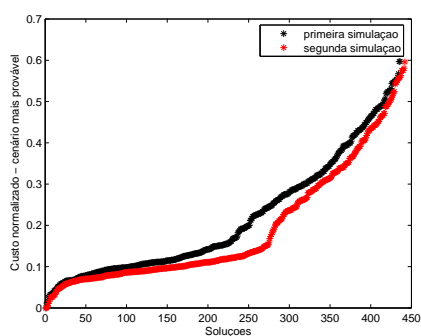
Pela tabela 4.10 pode-se observar os dados relativos aos gastos com avaliação de função objetivo em cada etapa do algoritmo: o número de soluções obtidas para formar o conjunto de soluções, o número de soluções não-dominadas na análise de sensibilidade multiobjetivo e o número de soluções com taxa de infactibilidade menor ou igual à 20%. Por esta tabela, o algoritmo clonal, embora apresente o menor custo com avaliação de função objetivo (já que não necessita de avaliações para a busca local) e apresente um

Tabela 4.8: Média da função objetivo para o cenário mais provável

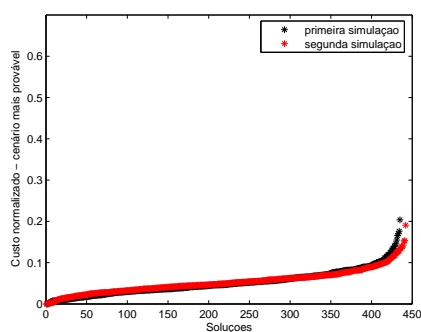
Algoritmo	Média
Clonal(1ª sim.)	1.343.407,28
Clonal(2ª sim.)	1.315.558,07
AM1(1ª sim.)	1.162.033,38
AM1(2ª sim.)	1.163.890,88
AM2(1ª sim.)	1.216.821,72
AM2(2ª sim.)	1.220.913,98
AM3(1ª sim.)	1.161.129,66
AM3(2ª sim.)	1.162.023,34

onde:

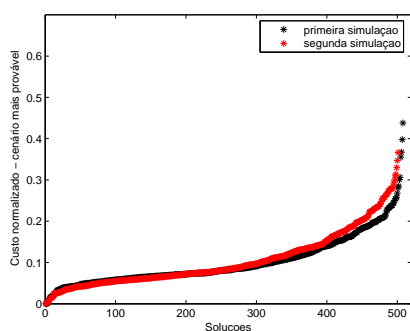
Média: média do valor da função objetivo para o cenário mais provável do conjunto de soluções obtido



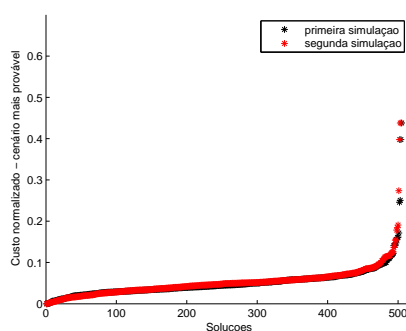
(a) Soluções obtidas pelo AC



(b) Soluções obtidas pelo AM1



(c) Soluções obtidas pelo AM2



(d) Soluções obtidas pelo AM3

Figura 4.27: Soluções encontradas nas execuções realizadas pelos algoritmos de otimização

conjunto de soluções de tamanho comparável aos demais algoritmos, para ele as soluções não-dominadas têm desempenho pior que as soluções dos outros algoritmos. Isto é observado quando agrupamos em um mesmo conjunto as soluções obtidas pelo AC e as soluções dos outros algoritmos. O conjunto de soluções não-dominadas é predominantemente formado por soluções pertencentes aos algoritmos meméticos, conforme pode ser observado na tabela 4.9. Por esta mesma tabela pode-se observar que para as duas simulações o AM3 foi o que mais contribuiu para a formação do conjunto de soluções não-dominadas.

Tabela 4.9: Soluções não-dominadas resultantes do agrupamento entre respostas dos algoritmos

$Clonal \cup AM1 \cup AM2 \cup AM3$	NSND	NSNDAM3	NSNDAM2	NSNDAM1	NSNDCL
1ª simulação	73	28 de 35	18 de 25	22 de 49	5 de 6
2ª simulação	74	35 de 41	7 de 31	25 de 41	7 de 9

onde:

NSND: número de soluções não-dominadas do conjunto formado pela união entre as soluções dos algoritmos: clonal, AM1, AM2 e AM3

NSNDCL: número de soluções não-dominadas que são provenientes do algoritmo clonal

NSNDAM1: número de soluções não-dominadas que são provenientes do algoritmo AM1

NSNDAM2: número de soluções não-dominadas que são provenientes do algoritmo AM2

NSNDAM3: número de soluções não-dominadas que são provenientes do algoritmo AM3

Para analisar os resultados de cada um dos algoritmos foi plotado em um único gráfico as respostas de cada algoritmo para cada simulação, sendo que estas soluções foram ordenadas em ordem crescente de valor da função objetivo. Foi feita uma normalização para o eixo das abscissas, uma vez que cada algoritmo retorna um número de soluções ótimas diferente. Também foi feito um outro gráfico em que as soluções de cada algoritmo foi ordenada no eixo das abscissas de acordo com a distância de cada solução ao ótimo para o cenário mais provável. Os gráficos para a primeira simulação de cada um dos algoritmos utilizados neste trabalho são apresentados na figura 4.28, onde a figura 4.28(a) apresenta as soluções ordenadas em ordem crescente de valor da

função objetivo e a figura 4.28(b) apresenta as soluções ordenadas de acordo com a distância em relação ao ótimo. Os gráficos para a segunda simulação realizada para cada algoritmo são apresentados nas figuras 4.28(c) e 4.28(d).

É possível ver pelos gráficos 4.28(a) e 4.28(c) que o algoritmo clonal é o algoritmo que apresenta o pior resultado, pois ao ordenar as soluções obtidas em valor crescente de valor da função objetivo, suas soluções tanto para a primeira quanto para a segunda simulação foram as que apresentaram os maiores valores. Pode-se observar que os algoritmos meméticos tendem a agrupar as soluções encontradas mais próximas à rede ótima do cenário mais provável, criando aglomerações em regiões formadas por faixas de distâncias. Isto indica uma característica de busca local. Por outro lado, o algoritmo clonal tende a espalhar as soluções por todas as direções.

Tabela 4.10: Dados das execuções realizadas

Algoritmo	<i>NAFO</i>	<i>NAC</i>	<i>NA MBL1</i>	<i>NA MBL2</i>	<i>NSE</i>	<i>NSPO</i>	<i>NSA</i>	<i>PVO</i>	<i>PVC</i>
Clonal (1ª sim.)	81.210	81.210	-	-	436	39	6	7,87	75,58
Clonal (2ª sim.)	80.847	80.847	-	-	442	46	9	5,46	78,97
AM1 (1ª sim.)	213.159	81.210	131.949	-	435	49	8	1,98	98,34
AM1 (2ª sim.)	203.970	80.847	123.123	-	442	41	6	1,29	73,69
AM2 (1ª sim.)	235.802	159.528	-	76.274	508	25	12	0,66	73,28
AM2 (2ª sim.)	225.078	152.748	-	72.330	501	31	7	0,19	77,54
AM3 (1ª sim.)	341.110	159.528	105.308	76.274	440	35	10	0,19	68,17
AM3 (2ª sim.)	323.152	153.313	98.831	71.008	503	41	10	0,19	72,74

onde:

NAFO: é o número de avaliações total da função objetivo realizadas pelo algoritmo;

NAC: é o número de avaliações realizadas pelo algoritmo clonal;

NA MBL1: é o número de avaliações da função objetivo realizadas pelo MBL1;

NA MBL2: é o número de avaliações da função objetivo realizadas pelo MBL2;

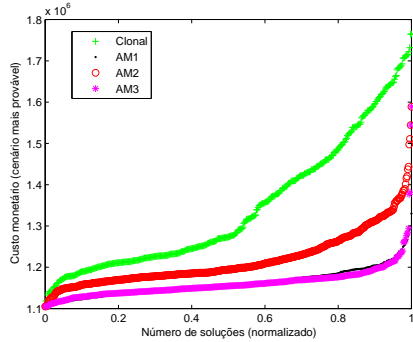
NSE: é o número de soluções ótimas obtidas;

NSPO: é o número de soluções não-dominadas obtidas pela análise de sensibilidade multiobjetivo;

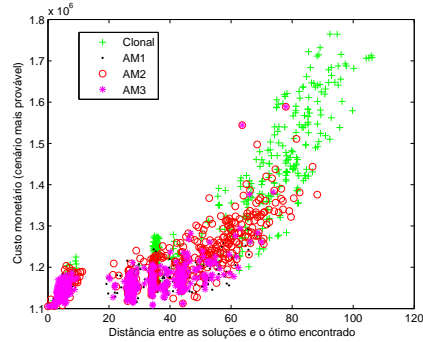
NSA: é o número de soluções não-dominadas que são opções de implementação (taxa de infactibilidade menor ou igual à 20%);

PVO: é a porcentagem do valor em que a função objetivo da rede escolhida (priorizando custo para o cenário médio) é maior que a rede ótima para o cenário mais provável;

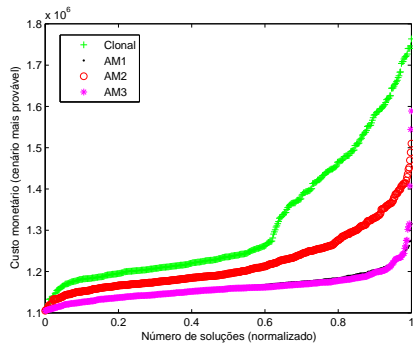
PVC: é a porcentagem do valor médio por falhas em que a rede escolhida (priorizando confiabilidade) representa sobre o valor do custo médio por falhas da rede ótima para o cenário mais provável.



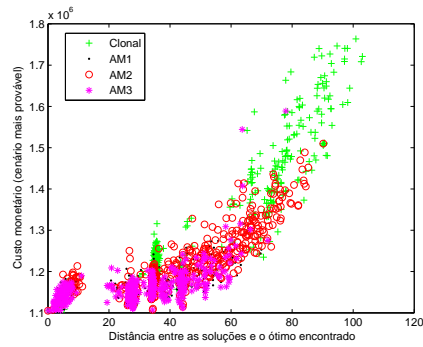
(a) Ótimos encontrados pelos algoritmos de otimização ordenados por valor de função objetivo, resultados para a primeira simulação



(b) Ótimos encontrados pelos algoritmos de otimização ordenados por distância em relação ao ótimo para o cenário mais provável, resultados para a primeira simulação



(c) Ótimos encontrados pelos algoritmos de otimização ordenados por valor de função objetivo, resultados para a segunda simulação



(d) Ótimos encontrados pelos algoritmos de otimização ordenados por distância em relação ao ótimo para o cenário mais provável, resultados para a segunda simulação

Figura 4.28: Comparação dos resultados obtidos pelos algoritmos de otimização

4.2 O Sistema de 100 nós

No sistema de 100 nós não existe nenhuma conexão inicial, ou seja, todos os 100 nós do sistema serão conectados por novas conexões e utilizando os mesmos condutores do sistema de 21 nós (tabela 4.2).

Assim como para o sistema de 21 nós, foi utilizada a técnica *Controlled-Greedy Encoding* para reduzir a dimensão do problema, com $mxv = 6$ e $mnv = 4$, resultando

em 291 variáveis como apresentado na figura 4.29. O horizonte de tempo considerado é de 10 anos e os dados das distribuições de probabilidade para o crescimento da carga e para a variação do preço de energia são apresentados na tabela 4.1.

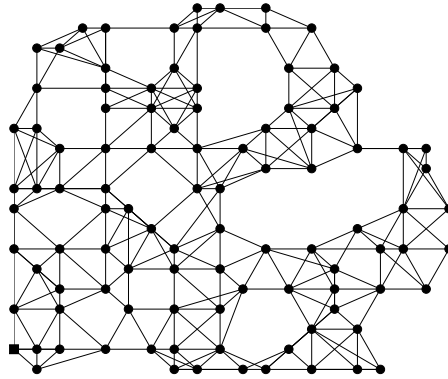


Figura 4.29: Sistema de 100 nós - Codificação por *Controlled-Greedy Encoding*

Para o caso de 100 nós foram executadas apenas uma simulação para o algoritmo clonal e uma simulação para o algoritmo AM3. O objetivo é ilustrar a melhoria que se obtém com a implantação de um algoritmo memético para um projeto de redes de dimensão mais elevada. O algoritmo memético AM3 foi escolhido por ser o algoritmo que apresentou os melhores resultados para as simulações feitas para o problema de 21 nós. Os parâmetros dos algoritmos foram mantidos exatamente os mesmos da simulação para o caso de 21 nós.

4.2.1 Resultados Obtidos via Algoritmo Clonal

Foram encontradas 128 soluções ótimas para o problema de 100 nós com a execução do Algoritmo Clonal, sendo para isso necessárias 123.790 avaliações da função objetivo. Estas soluções estão apresentadas em ordem crescente de valor da função objetivo na figura 4.30.

Das 128 soluções ótimas, após serem analisadas pela análise de sensibilidade multi-objetivo, restaram 23 soluções não-dominadas, apresentadas na tabela A.9 do apêndice

Tabela 4.11: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o caso de 100 nós

Solução	f_1	f_2	f_3	f_4
Algoritmo Clonal				
Ótimo para o cenário mais provável				
$1cl^{100}$	6801461,28	0,6000	6931302,58	6652020,82
Soluções Aplicáveis				
$32cl^{100}$	7417946,10	0,0000	7543752,90	6400007,58
$33cl^{100}$	7420977,83	0,0000	7546708,10	6399909,34
$38cl^{100}$	7460925,59	0,1900	7584209,46	6389456,97
$46cl^{100}$	7500456,10	0,1900	7621473,78	6257700,08
$47cl^{100}$	7530973,29	0,1900	7651652,13	6249155,86
Algoritmo AM3				
Ótimo para o cenário mais provável				
$1am3^{100}$	6698459,40	1,0000	-	-
Soluções Aplicáveis				
$21am3^{100}$	6698731,76	0,0000	6828316,03	6642755,41
$31am3^{100}$	6699192,38	0,0000	6825814,05	6493372,95
$41am3^{100}$	6702132,92	0,0000	6828737,81	6493364,67
$51am3^{100}$	6717052,54	0,0000	6843034,52	6423621,38
$81am3^{100}$	6739721,15	0,0000	6876730,73	6071747,97
$101am3^{100}$	6742099,90	0,0000	6869924,64	5642070,78
$191am3^{100}$	6749497,29	0,0000	6883501,65	5454545,48
$361am3^{100}$	6772817,52	0,0000	6909785,08	5324966,74
$531am3^{100}$	6786578,27	0,0000	6926967,14	5310211,41
$591am3^{100}$	6794662,21	0,0000	6932699,26	5205106,96
$861am3^{100}$	6837425,87	0,2000	6978197,65	5120800,77
$1861am3^{100}$	6954127,44	0,0100	7088300,10	5050596,76

A. A solução $1cl^{100}$ é a rede ótima para o cenário mais provável, ela é apresentada na figura 4.31. No entanto, ela apresenta uma alta taxa de inactibilidade (60,00%) indicando sua baixa robustez para cenários diferentes ao cenário mais provável. Das soluções pertencentes ao conjunto Pareto-ótimo, 5 soluções apresentam taxa de inactibilidade não superior a 20% ($32cl^{100}$, $33cl^{100}$, $38cl^{100}$, $46cl^{100}$ e $47cl^{100}$). A tabela 4.11 apresenta estas soluções. A solução escolhida, na situação em que a prioridade do projeto é uma rede robusta e com custo para o cenário mais provável mais próximo da rede ótima, deve ser a solução $32cl^{100}$, figura 4.32. Ela apresenta um custo para o cenário mais provável de aproximadamente 1,00% superior ao custo da rede ótima e é factível para todos os cenários simulados pela análise de sensibilidade multiobjetivo. Caso o projeto priorize a confiabilidade da rede, ou seja, dê preferência para a rede que apresente o menor custo médio por falhas, a rede a ser escolhida é a rede $47cl^{100}$, figura 4.33.

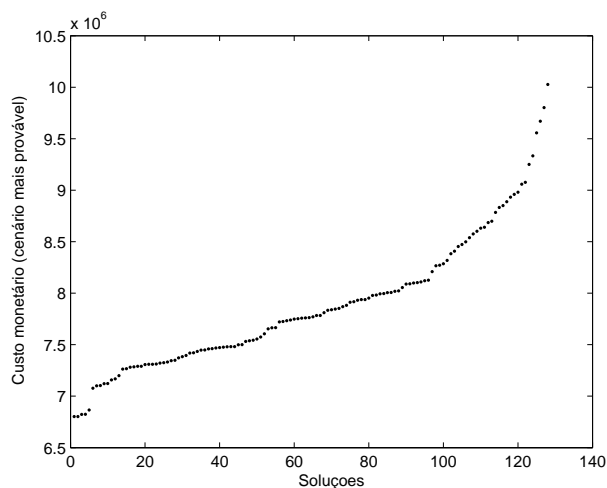


Figura 4.30: Redes obtidas pela otimização via Algoritmo Clonal para o caso de 100 nós

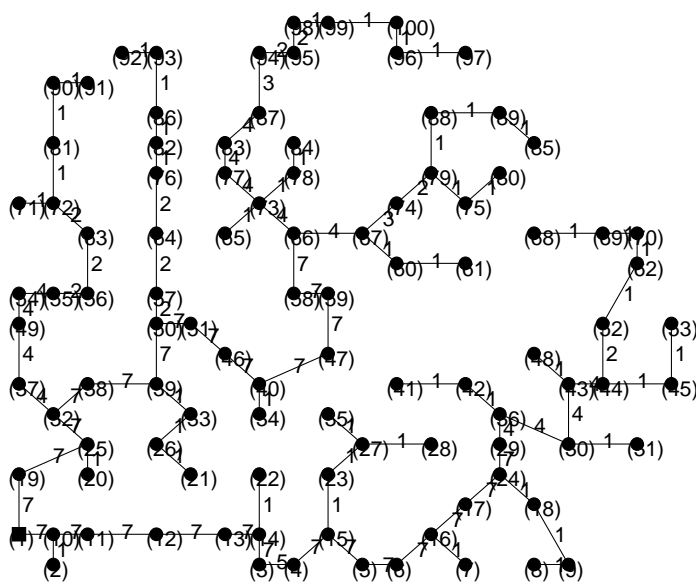


Figura 4.31: Rede ótima para o cenário mais provável para o caso de 100 nós - Algoritmo Clonal

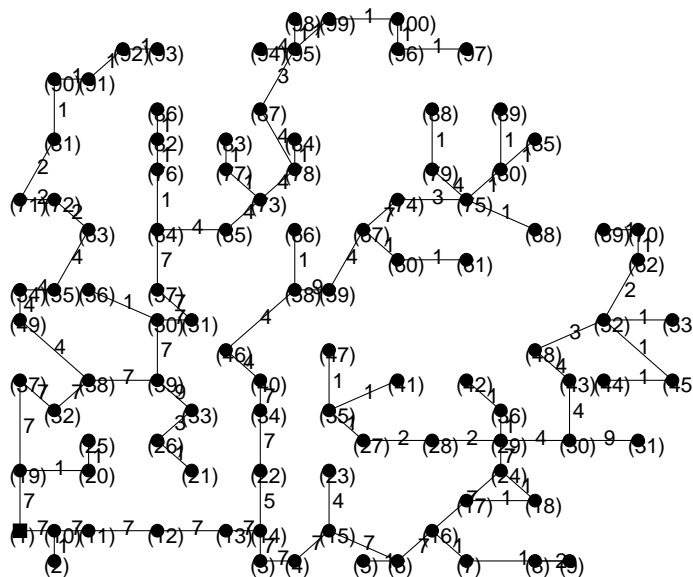


Figura 4.32: Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando custo para o cenário médio)- Algoritmo Clonal

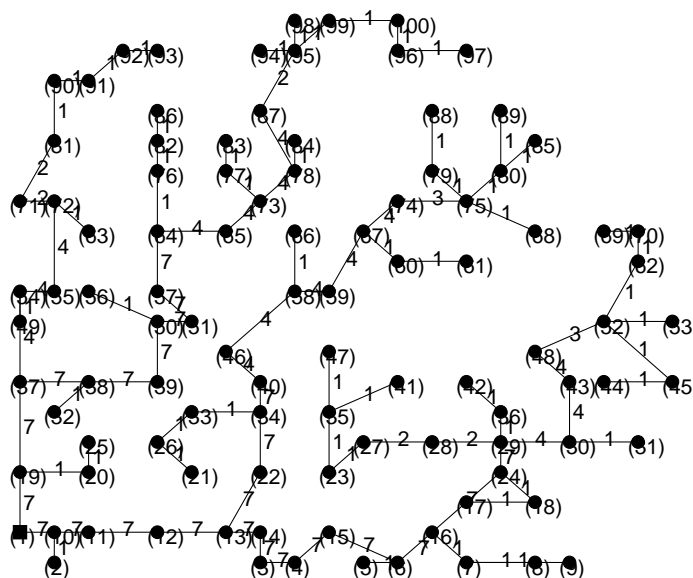


Figura 4.33: Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando confiabilidade) - Algoritmo Clonal

4.2.2 Resultados Obtidos via AM3

Foram encontradas 266 soluções ótimas nesta execução do *AM3*, sendo para isso necessárias 487.290 avaliações da função objetivo (246.072 avaliações gastas pelo algoritmo clonal, 216.024 avaliações gastas pelo MBL1 e 25.194 avaliações gastas pelo MBL2). Estas soluções estão apresentadas em ordem crescente de valor da função objetivo na figura 4.34.

Das 266 soluções ótimas, após serem analisadas pela análise de sensibilidade multi-objetivo, restaram 15 soluções não-dominadas, apresentadas na tabela A.10 do apêndice A. A solução $11am3^{100}$, apresentada na figura 4.35, é a rede ótima para o cenário mais provável. No entanto ela não foi factível em nenhum dos cenários simulados. Isto indica sua alta sensibilidade às condições de operação para as quais ela foi projetada. Das soluções pertencentes ao conjunto Pareto-ótimo, 12 soluções apresentam taxa de infactibilidade não superior a 20% ($21am3^{100}$, $31am3^{100}$, $41am3^{100}$, $51am3^{100}$, $81am3^{100}$, $101am3^{100}$, $191am3^{100}$, $361am3^{100}$, $531am3^{100}$, $591am3^{100}$, $861am3^{100}$ e $1861am3^{100}$). A tabela 4.11 mostra estas soluções. A solução escolhida quando a prioridade do projeto é uma rede robusta e com custo para o cenário médio o menor possível deve ser a rede a $21am3^{100}$, figura 4.37. Ela apresenta um custo para o cenário mais provável de apenas 0,10% superior ao custo da rede ótima e é robusta o suficiente para ser factível para todos os cenários simulados pela análise de sensibilidade multiobjetivo. Para a priorização da confiabilidade da rede, a escolha fica pela rede $1861am3^{100}$, que apresenta o menor custo médio para falhas nos cenários em que ela foi factível.

4.2.3 Comparação dos Resultados Obtidos para o Caso de 100 nós

Pelos resultados obtidos para o sistema de 100 nós pelos algoritmos clonal e *AM3*, nota-se que o *AM3* apresentou os melhores resultados. O ótimo encontrado por este algoritmo foi 2% mais barato que o encontrado pelo *AC*. Isto demonstra que o método de

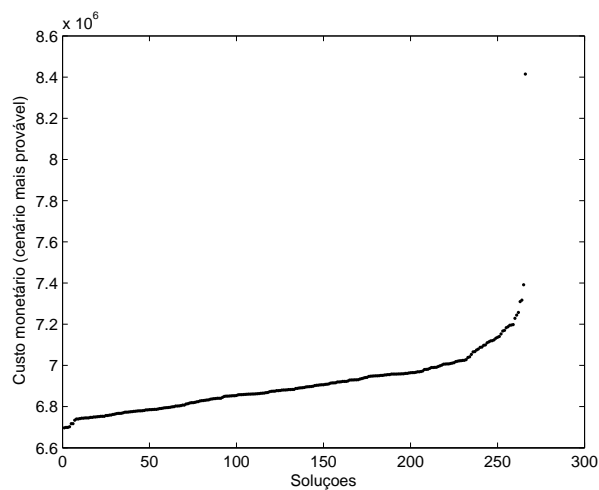


Figura 4.34: Redes obtidas pela otimização via AM3 para o caso de 100 nós

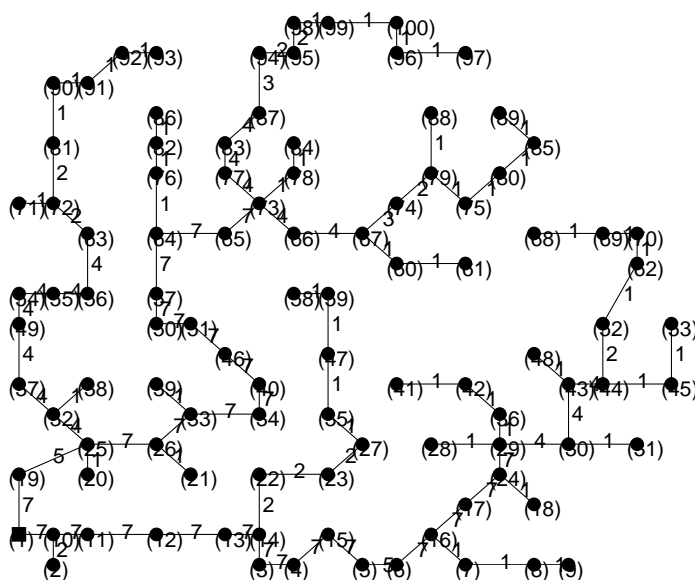


Figura 4.35: Rede ótima para o cenário mais provável para o caso de 100 nós - AM3

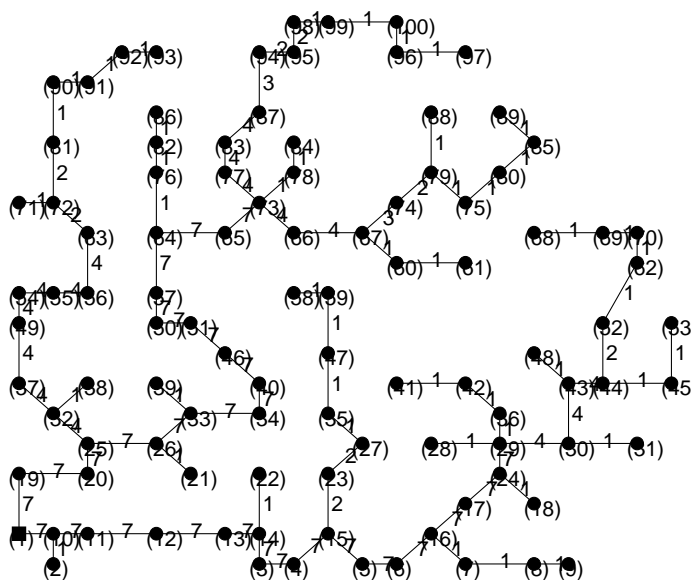


Figura 4.36: Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando custo para o cenário médio)- AM3

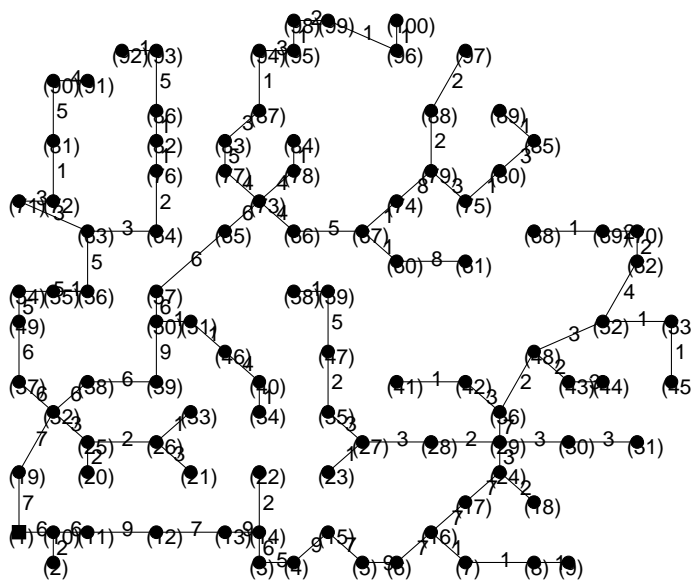
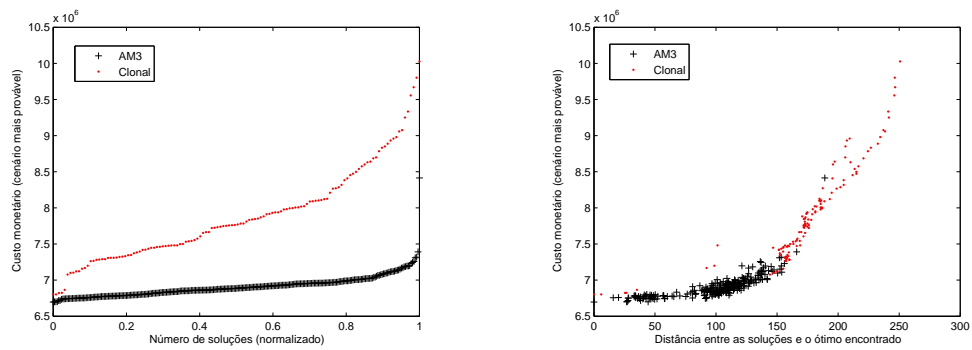


Figura 4.37: Rede escolhida para ser aplicada ao projeto do caso de 100 nós (priorizando confiabilidade) - AM3

busca local empregado neste algoritmo é capaz de melhorar a convergência do algoritmo, as soluções encontradas apresentam avaliações melhores do que as encontradas pelo Algoritmo Clonal e ainda, o número de soluções disponíveis para a escolha do projetista é maior e com qualidade melhor. A rede escolhida priorizando o custo no cenário médio do AM3 é 10% mais barata que a rede escolhida pela simulação do AC. Na priorização da confiabilidade, a rede escolhida na simulação do AM3 tem custo médio por falhas 20% menor que a rede escolhida pela simulação do AC.

A média do custo para o cenário mais provável das soluções obtidas pelo AC é de \$7.886.427,7830 enquanto que o custo médio para o AM3 é \$6.909.571,5195. Pode-se notar então que o AM3 tende a encontrar soluções cujo custo para o cenário médio é mais barato que para o AC. O número de avaliações de função objetivo gasto pelo AM3 é bem maior que o gasto pelo AC. Isto é devido aos gastos com as avaliações de função objetivo requerido pelo método de busca local presente no AM3. Quando são agrupadas em um mesmo conjunto as soluções encontradas pelo AC e pelo AM3 e é feita a análise de sensibilidade multiobjetivo, o conjunto de soluções não-dominadas é formado por 70% de soluções provenientes do AM3.

A figura 4.38(a) mostra como as soluções obtidas pelo AM3 têm, em geral, custo para o cenário mais provável bem menor que as soluções obtidas pelo AC. Na figura 4.38(b) pode-se notar que o AM3, da mesma maneira que para o sistema de 21 nós, tende a concentrar as suas soluções em regiões mais próximas à solução ótima, formando regiões em distâncias definidas. Já o AC tende a espalhar suas soluções por todo o espaço, sem demonstrar uma característica de busca local.



(a) Ótimos encontrados pelos algoritmos de otimização ordenados por valor de função objetivo

(b) Ótimos encontrados pelos algoritmos de otimização ordenados por distância em relação ao ótimo para o cenário mais provável

Figura 4.38: Comparação dos resultados obtidos pelos algoritmos de otimização para o caso de 100 nós

Capítulo 5

Conclusões

A utilização de algoritmos meméticos, inspirados nos sistemas imunológicos artificiais e nos métodos de busca local que foram desenvolvidos neste trabalho, permite obter um conjunto de soluções ótimas e sub-ótimas para o projeto de redes de distribuição de energia elétrica. Pelo emprego da análise de sensibilidade multiobjetivo é possível tratar as incertezas inerentes ao projeto, permitindo ao projetista a escolha de soluções que melhor se aplicam às suas necessidades, com garantia de robustez às variações na configuração da carga e do preço de energia no decorrer do tempo. A aplicação dos algoritmos meméticos permite ao projetista obter redes que são robustas às incertezas do projeto e que apresentam custos pouco superiores aos custos das redes ótimas que são menos robustas. A rede ótima, para os casos simulados, apresenta uma taxa de infactibilidade que a torna muito sensível às incertezas do projeto. Pequenos distúrbios tornam esta rede infactível, e ela não é capaz de atender à configuração operacional do sistema.

Os métodos de busca local desenvolvidos são baseados na geração de redes à distâncias pré-definidas, o que é possível graças à representação das redes no espaço contínuo, fundamentado pela métrica *T-norm*, permitindo trabalhar as redes com as propriedades deste espaço. Com o uso desta métrica é então possível a determinação de um conceito de busca local, devido uma definição consistente de distância entre redes. Estes métodos de busca local tornaram possível a implementação dos algoritmos meméticos. Os

resultados obtidos por estes algoritmos mostram que os mesmos possuem a capacidade de refinar consideravelmente os ótimos locais, permitindo o mapeamento de alternativas de projeto mais adequadas do ponto de vista prático.

A aplicação do algoritmo memético proposto em um sistema de 100 nós mostra que o mesmo é capaz de lidar com problemas muito complexos, fornecendo soluções de alta qualidade. As soluções obtidas pelo *AM3* são, em muito, superiores às soluções obtidas pelo *AC*, tanto qualitativamente quanto quantitativamente.

A abordagem proposta permite o uso de diferentes critérios de decisão. Neste trabalho foi priorizado a escolha de redes robustas e de baixo custo. No entanto, outros critérios tais quais confiabilidade, queda de tensão, etc. podem também ser priorizados, sem a exigência de adaptações na abordagem proposta.

A obtenção de redes robustas que é propiciada pelo algoritmo proposto reduz consideravelmente a possibilidade de se ter um sistema sub-dimensionado. Isto, na maioria das vezes, evita a necessidade de novos investimentos em re-expansão do sistema.

Aqui o conceito de busca local e os métodos de busca local desenvolvidos foram utilizados no projeto de redes de distribuição de energia elétrica. No entanto, nada impede que os conceitos desenvolvidos possam ser expandidos e utilizados em outros problemas cuja representação esteja no espaço de redes. Exemplos desta possibilidade de utilização seriam as redes de distribuição de água, redes de telecomunicações, redes de transporte, ou outros problemas em que as variáveis possam ser representadas por redes.

Capítulo 6

Propostas de Continuidade

Na seqüência são apresentadas possíveis continuidades deste trabalho:

1) Estudo de outras alternativas para a caracterização de vizinhança em espaço de redes

Com a utilização das propriedades adquiridas do espaço contínuo quando as redes são representadas neste espaço, estudar outras possibilidade de definição de vizinhança em espaço de redes, buscando dar mais consistência à definição proposta. Assim, os métodos de busca local passam a trabalhar sob uma região melhor definida possibilitando a obtenção de melhores resultados;

2) Aplicar/adaptar os algoritmos meméticos propostos neste trabalho para a solução de outros problemas:

Utilizar os Algoritmos Meméticos desenvolvidos, principalmente o AM3, em outros problemas de redes, tais como:

- projeto de redes de distribuição de água e de gás;
- projeto de redes de telecomunicações e de informática.

Será necessário para a aplicação dos algoritmos meméticos nestes problemas apenas a adaptação quanto à representação das variáveis e a definição de RL ;

3) Estudo da melhor configuração do algoritmo memético para o projeto de redes de distribuição

Neste trabalho, com os algoritmos meméticos desenvolvidos, foram estudados apenas os efeitos decorrentes do método de busca local empregado e da periodicidade com que os métodos de busca local são usados. Seria interessante um estudo do comportamento do algoritmo memético frente a outros ajustes, como por exemplo, definir sob que parcela da população será aplicado os métodos de busca local e a intensidade da busca local, priorizando a redução do custo computacional de avaliação da função objetivo.

4) Utilização da metodologia utilizada para promover o *scheduling* da expansão do sistema de distribuição

Os algoritmos meméticos e a análise de sensibilidade multiobjetivo permitiriam a obtenção de uma rede alvo para a programação dinâmica. A programação dinâmica por sua vez seria responsável pelo *scheduling* da expansão do sistema de distribuição de energia.

5) Otimização multiobjetivo de redes de distribuição de energia elétrica com incertezas na evolução da carga

Utilizar a otimização multiobjetivo para a obtenção das soluções candidatas para o tratamento de incertezas pela análise de sensibilidade multiobjetivo. Ao invés de um cenário médio para a otimização do projeto de redes de distribuição, as funções objetivo da otimização multiobjetivo seriam avaliadas por uma análise de sensibilidade feita a cada iteração do algoritmo para um número reduzido de cenários.

Referências Bibliográficas

- Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms and Applications*. Prentice Hall, 1st edition.
- Bazaraa, M. S., Jarvis, J. J., and Sherali, H. D. (1991). *Linear Programming and Network Flows*. Wiley, 2nd edition.
- Bondy, J. A. and Murty, U. S. R. (1976). *Graph Theory with Applications*. MacMillan.
- Carrano, E. G. (2007). *Algoritmos Evolucionários Eficientes para Otimização de Redes*. PhD thesis, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil.
- Carrano, E. G., Guimaraes, F. G., Takahashi, R. H. C., Neto, O. M., and Campelo, F. (2007a). Electric distribution network expansion under load evolution uncertainty using an immune system inspired algorithm. *IEEE Transactions on Power Systems*, 22:851–861.
- Carrano, E. G., Soares, L. A. E., Takahashi, R. H. C., Saldanha, R. R., and Neto, O. M. (2006). Electric distribution multiobjective network design using a problem-specific genetic algorithm. *IEEE Transactions on Power Delivery*, 21:995–1005.
- Carrano, E. G., Takahashi, R. H. C., Cardoso, E. P., Saldanha, R. R., and Neto, O. M. (2005). Optimal substation location and energy distribution network design using a hybrid GA-BFGS algorithm. *IEE Proceedings on Generation, Transmission and Distribution*, 152(6):919–926.
- Carrano, E. G., Takahashi, R. H. C., Fonseca, C. M., and Neto, O. M. (2007b). Neto. non-linear network topology optimization - an embedding vector space approach. *Submitted to IEEE Transactions on Evolutionary Computation*.
- Carvalho, P. M. S., Ferreira, L. A. F. M., Lobo, F. G., and Barruncho, L. M. F. (1998). Optimal distribution network expansion planning under uncertainty by evolutionary decision convergence. *Electric Power and Energy Systems*, 20(2):125–129.
- Cayley, A. (1989). A theorem on trees. *Quart. Journal of Mathematics*, 23:376–378.

- Ching-Tzong, S. and Guor-Rurng, L. (2002). Reliability design of distribution systems using modified genetic algorithms. *Electrical Power and System Research*, 60(3):201–206.
- Cossi, A., Romero, R., and Mantovani, J. (2005). Planning of secondary distribution circuits through evolutionary algorithms. *IEEE Trans. Power Delivery*, 20(1):205–213.
- D. Costa, A. H. and Dubouis, O. (1995). Embedding of a sequential procedure within an evolutionary algorithm for coloring problems in graphs. *Journal of Heuristics*, 1:105–128.
- Dasgupta, D., editor (1998). *Artificial Immune Systems and their Applications*. Springer, Berlin, Germany.
- Dasgupta, D., Ji, Z., and Gonzalez, F. (1998). A new genetic local search algorithm for graph coloring. In *V Parallel Problem Solving From Nature*, pages 745–754, Amsterdam, The Netherlands.
- Dasgupta, D., Ji, Z., and Gonzalez, F. (2003). Artificial immune system (AIS) research in the last five years. In *Proc. IEEE Congress on Evolutionary Computation*, pages 123–130, Canberra, Australia.
- Davis, L., editor (1991). *Handbook of Genetic Algorithms*. Van Nostrand, New York.
- de Castro, L. N. (2001). *Engenharia Imunológica: Desenvolvimento e Aplicação de Ferramentas Computacionais Inspiradas em Sistemas Imunológicos Artificiais*. PhD thesis, Campinas, Brazil.
- de Castro, L. N. and Timis, J. (2002). *Artificial Immune Systems: A New Computational Intelligence Paradigm*. Springer-Verlag, Berlin, Germany.
- de Castro, L. N. and Timmis, J. (2002). An artificial immune network for multimodal function optimization. In *Proc. IEEE Congress on Evolutionary Computation*, pages 699–704.
- de Castro, L. N. and Timmis, J. (2003). Artificial immune systems as a novel soft computing paradigm. *Soft Computing*, 7(8):526–544.
- de Castro, L. N. and Von Zuben, F. J. (2002). Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation*, 6(3):239–251.
- Dijkstra, E. W. (1959). A note on two problems in connection with graphs. *Numerische Mathematik*, 1:269–271.

- Dorigo, M., Di Caro, G., and Gambardella, L. M. (1999). Ant algorithms for discrete optimization. *Artificial Life*, 5(3):137–172.
- Dorigo, M. and Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1):53–66.
- Dorigo, M., Maniezzo, V., and Colorni, A. (1996). The ant system: Optimization by a colony of cooperating agents. *IEEE Transaction on Systems, Man, and Cybernetics*, 26(1):1–13.
- Even, S. (1993). *Genetic hybrids for the quadratic assignment problem*. American Mathematics Society, Providence, RI.
- Fleurent, C. and Ferland, J. (1997). Genetic and hybrid algorithms for graph coloring. *Annual Operational Research*, 63:437–461.
- Freisleben, B. and Merz, P. (1996). A genetic local search algorithm for solving symmetric and asymmetric traveling salesman problems. In *Proc. IEEE Congress on Evolutionary Computation*, pages 616–621.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Professional.
- Goldberg, D. E. and Voessner, S. (1999). Optimizing global-local search hybrids. In *Proceedings of the genetic and evolutionary computation conference*, pages 220–228.
- Gómez, J. F., Khodr, H. M., De Oliveira, P. M., Ocque, L., Yusta, J. M., Villasana, R., and Urdaneta, A. J. (2004). Ant colony system algorithm for the planning of primary distribution circuits. *IEEE Transactions on Power Systems*, 19(2):996–1004.
- H. M. Voigt, W. Ebeling, I. R. and Schwefel, H. P. (1996). New genetic local search operators for the traveling salesman problem. In *Proc. 4th Conf. Parallel Problem Solving From Nature*, pages 890–900, Berlin, Germany.
- Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. The University of Michigan Press.
- J. Smith, W. H. and Krasnogor, N. (2004). Special issue on memetic algorithms. *Evolutionary Computation*, 12(3):273–353.
- J. Tang, M. H. L. and Ong, Y. S. (2006). Diversity-adaptive parallel memetic algorithm for solving large scale combinatorial optimization problems. *Soft Computing*, (11):873–888.

- J. Watson, C. Ross, V. E. J. D. J. B. C. G. D. W. and Howe, A. (1998). The traveling salesman problem, edge assembly crossover, and 2-opt. In *Proc. Parallel Problem Solving From Nature*, pages 823–832, Berlin, Germany.
- Jerne, N. K. (1973). The immune system. *Scientific American*, 229:52–60.
- Jong, K. A. D. (1993). Genetic algorithms are not function optimizers. *Foundations of Genetic Algorithms 2*, (4):5–17.
- Klein, J. (1990). *Immunology*. Blackwell Scientific Publications.
- Krasnogor, N. and Smith, J. (2000). A memetic algorithm with self-adaptive local search: TSP as a case study. In *Proc. Genetic Evolutionary Computation Conference*, pages 987–994.
- Krasnogor, N. and Smith, J. (2005). A tutorial for competent memetic algorithms: Model, taxonomy, and design issues. *IEEE Transactions on Evolutionary Computation*, 9(5):474–488.
- Lima, E. L. (1995). *Curso de Análise*. IMPA, Rio de Janeiro.
- Manno, I. (1999). *Introduction to the Monte-Carlo Method*. Akademiai Kiado, Budapest, Hungary.
- Merz, P. and Freisleben, B. (1997). Genetic local search for the TSP: New results. In *Proc. IEEE Congress on Evolutionary Computation*, pages 159–164.
- Merz, P. and Freisleben, B. (1999a). A comparison of memetic algorithms, tabu search, and ant colonies for the quadratic assignment problem. In *Proc. IEEE Congress on Evolutionary Computation*, pages 2063–2070.
- Merz, P. and Freisleben, B. (1999b). Genetic algorithms for binary quadratic programming. In *Proc. International Genetic Evolutionary Computation Conference*, pages 417–424.
- Miranda, V., Ranito, J. V., and Proença, L. M. (1994). Genetic algorithms in optimal multistage distribution network planning. *IEEE Transactions on Power Systems*, 9(4):1927–1933.
- Mitchell, M. (1996). *An introduction to genetic algorithms*. MIT Press.
- Moscato, J. P. (1989). *On evolution, search, optimization, GAs and martial arts: toward memetic algorithms*. PhD thesis, California Inst. Technol., Pasadena, CA.
- N. Krasnogor, B. Blackburne, E. B. and Hirst, J. (2002). Multimeme algorithms for protein structure prediction. In *Proc. Parallel Problem Solving From Nature*, pages 769–778.

- Nagata, Y. and Kobayashi, S. (1997). Edge assembly crossover: A high-power genetic algorithm for the traveling salesman problem. In *Proc. 7th Int. Conf. Genetic Algorithms*, pages 450–457.
- Narsingh, D. (1984). *Graph Theory with Applications to Engineering and Computer Science*. Prentice Hall, New Delhi, India.
- Ong, Y. S. and Keane, A. J. (2004). Meta-lamarckian learning in memetic algorithms. *IEEE Transactions on Evolutionary Computation*, 8(2):99–110.
- Papoulis, A. (1991). *Probability, random variables and stochastic processes*. McGraw-Hill Companies, 3rd edition.
- Pierre, S. (1993). Application of artificial intelligence techniques to computer network design. *Engineering Applications in Artificial Intelligence*, 6(5):465–472.
- R. Carr, W. Hart, N. K. E. B. J. H. and Smith, J. (2002). Alignment of protein structures with a memetic evolutionary algorithm. In *Proc. Genetic Evolutionary Computation Conference*, pages 1027–1034.
- Ramírez-Rosado, I. and Bernal-Agustín, J. (1998). Genetic algorithms applied to the design of large power distribution systems. *IEEE Transactions on Power Systems*, 13(3):696–702.
- Smith, D. K. and Walters, G. A. (2000). An evolutionary approach for finding optimal trees in undirected networks. *European Journal of Operational Research*, 120(3):593–602.
- Soares, L. A. E. (2001). Uma abordagem multi-critério no planejamento ótimo de sistemas de distribuição de energia. Master's thesis, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil.
- Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction*. MIT Press.
- Vanderbei, R. J. (2001). *Linear Programming: Foundations and Extensions*. Springer, 2nd edition.
- Vazquez, M. and Whitley, L. (2000). A hybrid genetic algorithm for the quadratic assignment problem. In *Proc. Genetic Evolutionary Computation*, pages 135–142.
- W. Hart, N. K. and Smith, J. (2005). *Recent Advances in Memetic Algorithms*. Springer-Verlag, Berlin, Germany.
- Wanner, E. F. (2006). *Operadores para Algoritmos Genéticos baseados em aproximações quadráticas de funções de variáveis contínuas*. PhD thesis, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil.

-
- Willis, H. L., Tram, H., Engel, M. V., and Finley, L. (1996). Selecting and applying distribution optimization methods. *IEEE Computer Applications in Power*, 9(1):12–17.
- Wilson, R. J. (1996). *Introduction to Graph Theory*. Prentice Hall, 4th edition.
- X. Wang, X. Z. G. and Ovaska, S. J. (2004). Artificial immune optimization methods and applications - a survey. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 3415–3420.

Apêndice A

Redes obtidas pelas simulações dos algoritmos de otimização

Neste apêndice são apresentadas as redes não-dominadas encontradas pela análise de sensibilidade multiobjetivo resultantes dos conjuntos soluções encontrados pelos algoritmos empregados.

A.1 Caso de 21 nós

A.1.1 Primeira simulação com o algoritmo clonal

Tabela A.1: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo - Primeira Simulação com o Algoritmo Clonal

Solução	f_1	f_2	f_3	f_4
1cl ^A	1105002,59	0,7190	1213229,95	1732476,16
3cl ^A	1119211,78	0,7371	1228755,56	1497171,46
12cl ^A	1152622,08	0,7160	1260826,46	1699076,27
15cl ^A	1160608,60	0,7296	1269147,52	1553934,16
17cl ^A	1163194,36	0,8262	1276171,63	1350065,23
27cl ^A	1177776,97	0,7900	1296431,26	1480149,15
28cl ^A	1178100,08	0,7900	1296740,49	1480138,47
32cl ^A	1178651,57	0,7900	1297770,24	1445136,04
34cl ^A	1179205,33	0,7900	1298372,90	1434435,59
37cl ^A	1181992,24	0,7960	1301435,52	1399985,76
43cl ^A	1187884,07	0,7975	1308250,71	1390700,51
48cl ^A	1190218,73	0,7749	1307257,37	1421526,42
50cl ^A	1192030,00	0,0060	1309338,10	1721028,42
64cl ^A	1200877,15	0,7900	1319902,21	1327169,01
66cl ^A	1201313,78	0,0060	1318427,57	1682633,80
70cl ^A	1203739,73	0,8716	1313298,47	1246744,02
85cl ^A	1210578,55	0,8006	1331222,13	1262519,43
108cl ^A	1216462,97	0,8670	1328106,66	1312649,18
110cl ^A	1216962,23	0,8670	1328457,00	1312483,10
151cl ^A	1232045,53	0,0060	1349438,71	1667692,85
184cl ^A	1250648,07	0,2462	1372871,89	1520726,47
203cl ^A	1263358,41	0,8821	1381756,67	1069515,84
210cl ^A	1269998,12	0,2462	1391715,81	1498667,90
211cl ^A	1270176,12	0,4531	1390773,53	1376882,89
214cl ^A	1271574,85	0,2462	1393043,60	1464864,32
220cl ^A	1274638,69	0,7281	1383728,64	1493095,93
223cl ^A	1276558,31	0,7296	1385641,39	1482987,75
225cl ^A	1277689,56	0,0498	1396528,59	1666312,46
228cl ^A	1282720,65	0,0045	1396296,91	1651553,65
247cl ^A	1322901,08	0,7552	1436861,58	1332182,63
264cl ^A	1358756,61	0,0120	1493389,08	1309433,98
277cl ^A	1377792,43	0,8731	1487479,25	1238066,76
279cl ^A	1380242,38	0,7673	1496911,95	1275214,41
295cl ^A	1408998,68	0,7416	1521567,23	1283640,89
311cl ^A	1427655,17	0,7371	1544142,83	854112,940
318cl ^A	1436590,91	0,8413	1543724,65	1241867,24
369cl ^A	1539730,47	0,2477	1671922,92	1010385,10
374cl ^A	1546474,33	0,5664	1671291,84	1307964,69
431cl ^A	1714716,87	0,6873	1849538,57	912578,16

A.1.2 Segunda simulação com o algoritmo clonal

Tabela A.2: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo - Segunda Simulação com o Algoritmo Clonal

Solução	f_1	f_2	f_3	f_4
$1cl^B$	1105002,59	0,6980	1212094,46	1727560,61
$3cl^B$	1108118,24	0,6968	1215152,37	1715784,64
$5cl^B$	1122674,79	0,6940	1229584,80	1728020,14
$11cl^B$	1139032,42	0,7180	1249348,96	1593457,61
$14cl^B$	1148783,18	0,7988	1258879,69	1576507,83
$20cl^B$	1161237,83	0,8328	1264291,11	1537200,18
$22cl^B$	1164744,78	0,6888	1269952,71	1695001,69
$23cl^B$	1165360,41	0,0032	1284710,10	1755828,40
$25cl^B$	1169509,22	0,7420	1283618,19	1504947,33
$28cl^B$	1170656,33	0,7444	1284903,42	1504535,52
$29cl^B$	1171313,46	0,7444	1285521,78	1504508,88
$53cl^B$	1183055,41	0,7508	1298189,26	1485302,25
$62cl^B$	1185793,59	0,7504	1300879,36	1474306,26
$67cl^B$	1187066,90	0,7308	1299377,77	1532949,92
$73cl^B$	1190424,37	0,7600	1306639,91	1353111,73
$74cl^B$	1190725,36	0,7596	1306884,21	1353083,84
$79cl^B$	1192432,68	0,7316	1303179,46	1498743,50
$151cl^B$	1212214,26	0,8340	1324750,33	1186858,51
$161cl^B$	1214544,56	0,8332	1327147,65	1177828,72
$217cl^B$	1234800,30	0,0348	1349825,97	1643260,24
$234cl^B$	1240181,09	0,8568	1356548,27	1137670,30
$238cl^B$	1242314,19	0,0032	1357270,98	1690033,59
$244cl^B$	1244076,49	0,8560	1360345,44	1174458,48
$262cl^B$	1259859,19	0,2436	1381345,37	1559857,41
$278cl^B$	1301154,97	0,0524	1417501,15	1634521,30
$279cl^B$	1307799,72	0,7252	1418805,16	1451886,25
$280cl^B$	1310670,63	0,7252	1421603,91	1441437,45
$294cl^B$	1356001,32	0,4656	1476734,13	1376989,77
$295cl^B$	1360290,57	0,2868	1488454,32	1221284,41
$297cl^B$	1363798,79	0,2844	1493539,31	1187968,96
$304cl^B$	1372059,59	0,7088	1481927,28	1290413,23
$305cl^B$	1372970,22	0,7060	1482117,92	1269680,03
$309cl^B$	1376102,08	0,2184	1512736,48	1468611,30
$312cl^B$	1390792,81	0,0100	1515806,79	1544348,15
$313cl^B$	1390858,19	0,2280	1517003,89	1059683,68
$315cl^B$	1393789,39	0,2680	1514656,00	1430401,27
$316cl^B$	1393902,18	0,6612	1497270,87	1168464,14
$317cl^B$	1394326,89	0,6536	1497097,26	1153654,44
$342cl^B$	1442831,67	0,7576	1559551,48	944365,99
$401cl^B$	1586709,06	1,0000	-	-
$405cl^B$	1594019,08	0,0032	1714264,60	1630519,64
$406cl^B$	1594767,67	0,0032	1714967,94	1630506,23
$411cl^B$	1609558,32	0,0012	1731990,43	2023950,20
$419cl^B$	1644176,91	0,0200	1781503,90	1368106,78

A.1.3 Primeira simulação com o AM1

Tabela A.3: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM1 - Primeira Simulação

Solução	f_1	f_2	f_3	f_4
1am1 ^A	1105002,59	0,6987	1211782,54	1726414,17
2am1 ^A	1105110,08	0,6983	1211865,27	1726500,13
3am1 ^A	1105189,03	0,6973	1211980,55	1726711,26
5am1 ^A	1108909,98	0,7017	1216000,72	1651417,64
6am1 ^A	1110773,74	0,6900	1215289,51	1676121,96
7am1 ^A	1110932,85	0,7046	1218538,92	1641688,62
8am1 ^A	1113188,42	0,7065	1221059,65	1606230,25
9am1 ^A	1113242,99	0,6924	1218445,51	1623331,44
11am1 ^A	1113354,03	0,6910	1218408,95	1623147,20
12am1 ^A	1113667,61	0,7056	1221554,29	1583488,43
14am1 ^A	1113833,75	0,7046	1221484,10	1595032,85
15am1 ^A	1114886,80	0,7051	1222605,09	1550164,39
20am1 ^A	1115758,95	0,6919	1220976,09	1575650,48
33am1 ^A	1120026,95	0,7109	1228982,15	1464304,62
38am1 ^A	1121353,99	0,6978	1228213,71	1535091,62
41am1 ^A	1122326,98	0,6973	1228999,60	1535116,01
42am1 ^A	1122428,41	0,6856	1226755,88	1640760,97
46am1 ^A	1122954,75	0,6978	1230007,47	1511906,32
57am1 ^A	1126893,34	0,0038	1246994,74	1736906,28
60am1 ^A	1127500,29	0,7177	1237522,61	1351966,25
63am1 ^A	1128437,71	0,7201	1238704,57	1351353,58
64am1 ^A	1128497,72	0,7056	1236184,97	1444302,66
74am1 ^A	1131420,08	0,6910	1236383,66	1612302,42
77am1 ^A	1132400,78	0,7187	1242772,76	1255326,20
82am1 ^A	1133810,35	0,0038	1255656,73	1615977,87
85am1 ^A	1135266,63	0,7094	1243629,49	1302208,42
95am1 ^A	1136752,37	0,0038	1259746,55	1602717,09
105am1 ^A	1138891,77	0,6953	1245390,46	1544713,04
109am1 ^A	1139598,91	0,2681	1259748,10	1482237,24
115am1 ^A	1140624,82	0,6856	1244956,52	1630023,11
119am1 ^A	1141073,09	0,7299	1253113,17	1099583,46
136am1 ^A	1143551,47	0,7007	1250560,28	1444828,11
140am1 ^A	1144185,25	0,6905	1248973,97	1605199,76
151am1 ^A	1145339,70	0,7936	1250645,94	1188167,78
169am1 ^A	1148301,69	0,7172	1258354,85	1287925,36
172am1 ^A	1148729,03	0,7065	1256824,73	1401659,17
178am1 ^A	1150551,12	0,6948	1257033,95	1527365,07
197am1 ^A	1153160,58	0,7065	1261079,54	1257890,10
203am1 ^A	1154635,37	0,7245	1265816,99	1191406,12
237am1 ^A	1160659,78	0,0038	1281530,18	1402110,99
238am1 ^A	1160711,33	0,1995	1278617,37	1450056,06
249am1 ^A	1162294,00	0,6944	1268052,52	1360458,82
253am1 ^A	1163118,16	0,6885	1268142,40	1468169,76
277am1 ^A	1166856,86	0,6914	1272160,23	1455985,07
289am1 ^A	1169469,86	0,7270	1280841,03	1170541,64
315am1 ^A	1176779,82	0,1946	1293143,01	1156908,62
319am1 ^A	1177479,62	0,7094	1286029,81	1126358,64
383am1 ^A	1199906,34	0,0121	1321506,58	1372324,45
430am1 ^A	1266126,28	0,0024	1378026,51	1697831,21

A.1.4 Segunda simulação com o AM1

Tabela A.4: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM1 - Segunda Simulação

Solução	f_1	f_2	f_3	f_4
1am1 ^B	1105002,59	0,7002	1211842,93	1726082,47
3am1 ^B	1106125,71	0,7002	1212826,49	1725955,10
6am1 ^B	1108118,24	0,6998	1214834,44	1714050,19
10am1 ^B	1110125,15	0,6897	1214528,15	1675048,84
11am1 ^B	1113315,08	0,7050	1221440,81	1593944,28
14am1 ^B	1116087,94	0,6941	1222037,00	1664750,26
18am1 ^B	1118207,52	0,6949	1224237,04	1559304,66
19am1 ^B	1119287,86	0,0028	1238269,06	1689470,33
21am1 ^B	1119563,68	0,7118	1228712,48	1519874,10
23am1 ^B	1120569,20	0,7138	1229916,95	1451417,00
24am1 ^B	1120570,38	0,6998	1227525,41	1543888,54
41am1 ^B	1127250,43	0,6861	1231373,13	1675462,03
42am1 ^B	1127351,21	0,6853	1231503,27	1675670,45
50am1 ^B	1129740,15	0,7030	1237622,71	1301367,13
52am1 ^B	1131550,28	0,6925	1236608,02	1665257,52
55am1 ^B	1132303,89	0,6873	1236553,17	1597834,82
69am1 ^B	1135206,44	0,6937	1240666,78	1569366,66
70am1 ^B	1135700,55	0,6941	1241686,74	1543720,90
71am1 ^B	1135723,42	0,7332	1247019,10	1242072,36
81am1 ^B	1137139,15	0,6949	1243187,37	1534112,56
87am1 ^B	1138333,66	0,6977	1245203,74	1510881,31
90am1 ^B	1139008,54	0,6949	1244969,11	1442687,87
92am1 ^B	1139233,23	0,7002	1246208,80	1385183,97
109am1 ^B	1142596,73	0,6961	1249286,75	1401567,89
120am1 ^B	1145207,17	0,6998	1252134,68	1400654,19
125am1 ^B	1145676,12	0,7098	1254357,54	1277830,54
129am1 ^B	1146245,49	0,6989	1252935,91	1337191,67
145am1 ^B	1149369,43	0,7106	1258237,44	1246028,80
168am1 ^B	1152791,46	0,7054	1261107,78	1248365,15
176am1 ^B	1153773,45	0,6941	1259560,56	1454716,94
218am1 ^B	1159218,27	0,0028	1280493,77	1646169,48
228am1 ^B	1160852,63	0,7030	1268621,39	1160767,15
236am1 ^B	1162656,87	0,1883	1281516,62	1425793,14
238am1 ^B	1162945,35	0,6925	1268424,01	1488870,89
251am1 ^B	1165036,26	0,0028	1285005,87	1307260,78
289am1 ^B	1171871,01	0,7364	1283995,57	1064814,01
306am1 ^B	1174307,91	0,6849	1278287,35	1623520,74
313am1 ^B	1175662,68	0,0024	1289143,81	1669485,96
333am1 ^B	1179056,52	0,2008	1301106,09	1146106,03
375am1 ^B	1192514,69	0,1923	1313375,67	1272270,20
418am1 ^B	1216950,34	0,7332	1328271,18	1059168,06

A.1.5 Primeira simulação com o AM2

Tabela A.5: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM2 - Primeira Simulação

Solução	f_1	f_2	f_3	f_4
1am2 ^A	1105002,59	0,6834	1211575,99	1725468,46
2am2 ^A	1105682,69	0,6834	1212162,15	1725354,16
3am2 ^A	1107839,11	0,6827	1214287,12	1714471,33
4am2 ^A	1112306,93	0,0030	1227355,08	1681936,29
9am2 ^A	1124497,94	0,0030	1243666,16	1650669,73
11am2 ^A	1127196,32	0,7090	1236996,10	1439193,87
23am2 ^A	1142963,50	0,6985	1251885,12	1555974,54
31am2 ^A	1158782,70	0,7195	1269720,20	1212786,36
58am2 ^A	1147689,13	0,7000	1257071,95	1407438,60
100am2 ^A	1162766,27	0,6909	1271208,35	1328175,04
129am2 ^A	1161912,09	0,1691	1277462,24	1239120,88
132am2 ^A	1148686,67	0,0030	1273377,85	1479660,94
184am2 ^A	1177383,91	0,7195	1288330,40	1208416,16
220am2 ^A	1192293,25	0,7924	1300287,09	1010272,02
242am2 ^A	1197230,52	0,1736	1313319,64	1123244,42
243am2 ^A	1185373,67	0,1804	1303592,70	1119295,34
280am2 ^A	1186571,31	0,7240	1298252,57	1037226,84
287am2 ^A	1180626,96	0,1864	1301321,41	1147916,84
314am2 ^A	1153280,40	0,6947	1262092,86	1436651,91
320am2 ^A	1219697,97	0,1518	1330957,37	1380485,03
374am2 ^A	1242324,99	0,0030	1365927,27	1327096,57
396am2 ^A	1284599,55	0,1503	1394309,08	1294269,42
412am2 ^A	1227816,33	0,0398	1350952,03	1452512,50
427am2 ^A	1323312,90	0,0030	1446609,21	1264496,49

A.1.6 Segunda simulação com o AM2

Tabela A.6: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM2 - Segunda Simulação

Solução	f_1	f_2	f_3	f_4
1am2 ^B	1105110,08	0,7275	1214945,46	1736571,84
2am2 ^B	1107128,93	0,0030	1226976,74	1732700,60
3am2 ^B	1107428,41	0,7368	1217083,38	1733682,05
4am2 ^B	1110149,12	0,0030	1229935,43	1688689,55
5am2 ^B	1114596,86	0,0030	1231258,66	1626432,71
7am2 ^B	1118406,02	0,7244	1227922,97	1686462,68
8am2 ^B	1120436,04	0,7523	1231743,70	1599567,20
10am2 ^B	1121035,50	0,0030	1239987,28	1617068,47
15am2 ^B	1133375,35	0,7585	1246859,87	1337885,65
19am2 ^B	1134578,77	0,7430	1245398,86	1499500,82
24am2 ^B	1138292,77	0,2476	1257642,11	1544475,68
35am2 ^B	1145090,61	0,0030	1269208,86	1412587,71
41am2 ^B	1148337,63	0,7523	1260166,98	1341357,28
42am2 ^B	1148969,78	0,7399	1258596,77	1352781,86
44am2 ^B	1150306,09	0,7523	1263094,94	1292685,37
62am2 ^B	1154324,63	0,0061	1280126,19	1310008,19
65am2 ^B	1156082,41	0,7182	1264421,65	1468003,26
100am2 ^B	1166027,84	0,7523	1276998,92	1257816,38
102am2 ^B	1166256,19	0,7616	1281740,31	1105161,40
103am2 ^B	1166577,90	0,7523	1278785,33	1218065,37
147am2 ^B	1173284,35	0,7492	1284224,87	1295576,75
247am2 ^B	1193190,26	0,7739	1310271,87	1062343,25
254am2 ^B	1195558,63	0,2291	1312601,73	1295656,04
296am2 ^B	1210356,16	0,7244	1323210,48	1277474,49
372am2 ^B	1253800,27	0,3839	1386854,53	1249942,27
391am2 ^B	1263083,07	0,3188	1381310,94	1144327,19
397am2 ^B	1270036,63	0,3312	1390722,35	1006396,10
488am2 ^B	1407436,65	0,2910	1528691,99	1290568,58
490am2 ^B	1412424,22	0,7492	1523614,12	1001390,40
491am2 ^B	1412697,06	0,3126	1538591,90	1271782,82
498am2 ^B	1452151,61	0,0000	1581686,49	1337900,41

A.1.7 Primeira simulação com o AM3

Tabela A.7: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM3 - Primeira Simulação

Solução	f_1	f_2	f_3	f_4
1am3 ^A	1105002,59	0,6870	1212457,09	1728357,39
2am3 ^A	1105105,93	0,6870	1212490,49	1728301,35
4am3 ^A	1105682,69	0,6870	1213036,28	1728241,43
6am3 ^A	1107128,93	0,0000	1218338,25	1690987,33
9am3 ^A	1112042,71	0,6912	1220613,80	1598424,39
10am3 ^A	1112148,35	0,0041	1226916,09	1679831,60
14am3 ^A	1113610,10	0,6787	1219797,38	1678336,59
15am3 ^A	1114091,43	0,6926	1223521,31	1586310,03
17am3 ^A	1115310,07	0,6940	1224842,43	1586193,73
25am3 ^A	1118350,83	0,6828	1225078,85	1634470,75
26am3 ^A	1119044,62	0,6954	1229107,20	1477055,23
28am3 ^A	1119741,45	0,6787	1226727,02	1536079,79
30am3 ^A	1120904,91	0,6787	1227582,60	1501180,39
32am3 ^A	1121419,86	0,6801	1228411,73	1471397,49
36am3 ^A	1124206,09	0,7093	1234623,83	1423397,64
44am3 ^A	1126906,60	0,6703	1231830,63	1677295,56
55am3 ^A	1128933,35	0,1835	1246263,44	1361452,22
58am3 ^A	1129445,71	0,6912	1238652,32	1440513,14
73am3 ^A	1132459,32	0,7204	1243894,78	1376751,31
75am3 ^A	1133558,40	0,6745	1239157,75	1568673,87
98am3 ^A	1136801,16	0,6981	1246065,83	1250160,68
100am3 ^A	1137058,75	0,6787	1243780,60	1364132,93
134am3 ^A	1140953,66	0,7246	1253534,64	1127395,77
157am3 ^A	1143376,76	0,0041	1262585,52	1537115,63
177am3 ^A	1145766,01	0,7134	1256576,76	1073539,19
205am3 ^A	1149365,63	0,6912	1258114,23	1338609,10
258am3 ^A	1155061,72	0,1822	1270195,86	1474318,20
270am3 ^A	1156196,06	0,2378	1274881,14	1292277,66
272am3 ^A	1156281,49	0,1863	1274559,37	1227706,85
330am3 ^A	1164934,62	0,0041	1285145,61	1524281,15
366am3 ^A	1171117,81	0,0041	1288173,90	1491441,84
372am3 ^A	1171643,55	0,0041	1292614,17	1366650,29
416am3 ^A	1179673,76	0,1905	1299788,16	1178235,63
442am3 ^A	1187843,72	0,7232	1300233,09	1066317,83
449am3 ^A	1191254,47	0,7065	1301263,45	1166627,37

A.1.8 Segunda simulação com o AM3

Tabela A.8: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o AM3 - Segunda Simulação

Solução	f_1	f_2	f_3	f_4
1am3 ^B	1105002,59	0,7263	1212362,05	1728567,63
2am3 ^B	1107128,93	0,0000	1218338,25	1690987,33
4am3 ^B	1105737,96	0,7249	1213259,10	1720123,08
5am3 ^B	1107732,58	0,7292	1215303,04	1675215,33
6am3 ^B	1107992,99	0,7306	1215326,32	1674140,31
7am3 ^B	1108840,35	0,7292	1216203,33	1652616,23
10am3 ^B	1109688,27	0,7136	1214713,74	1678051,89
13am3 ^B	1111275,43	0,7165	1216740,22	1677858,35
14am3 ^B	1112255,29	0,0056	1228124,88	1681960,26
15am3 ^B	1112415,26	0,0056	1228186,18	1681873,95
21am3 ^B	1115816,67	0,7362	1223666,52	1582586,11
26am3 ^B	1118452,05	0,7221	1224970,60	1546493,18
29am3 ^B	1120419,49	0,0056	1239097,77	1629585,68
30am3 ^B	1120570,52	0,7221	1227149,73	1502139,92
34am3 ^B	1121389,25	0,7447	1230232,87	1473012,53
37am3 ^B	1122647,31	0,7433	1231314,96	1397772,99
51am3 ^B	1125718,53	0,0056	1246283,87	1576847,77
53am3 ^B	1126363,42	0,7447	1235975,66	1332261,80
66am3 ^B	1128829,20	0,7348	1236707,54	1491920,28
69am3 ^B	1129557,38	0,7136	1234591,60	1633977,03
70am3 ^B	1129913,91	0,7136	1234759,07	1605004,98
79am3 ^B	1133026,27	0,7179	1238884,60	1576735,83
81am3 ^B	1133143,02	0,7122	1237823,06	1633771,92
84am3 ^B	1133939,81	0,7489	1245417,47	1299195,43
88am3 ^B	1134605,48	0,7249	1242134,37	1397147,02
91am3 ^B	1135058,57	0,7193	1241015,46	1534831,67
94am3 ^B	1135257,86	0,7503	1246623,35	1222032,63
103am3 ^B	1137096,43	0,0056	1255255,68	1528115,34
107am3 ^B	1137689,61	0,7503	1249004,02	1157208,43
122am3 ^B	1140356,86	0,2186	1261228,86	1342590,24
146am3 ^B	1142764,43	0,7447	1252333,30	1171458,30
147am3 ^B	1143164,03	0,7376	1250437,24	1276445,30
206am3 ^B	1152539,59	0,7150	1257955,03	1523336,92
250am3 ^B	1157984,19	0,7489	1269264,75	996366,56
260am3 ^B	1159509,93	0,1819	1273662,50	1298224,61
262am3 ^B	1159620,96	0,7447	1268852,41	1117864,07
277am3 ^B	1160587,58	0,2158	1280798,21	1132098,82
301am3 ^B	1162040,45	0,0056	1277881,62	1522326,59
356am3 ^B	1169855,94	0,0056	1288084,76	1502394,07
370am3 ^B	1171879,51	0,0056	1294773,39	1257370,06
486am3 ^B	1233220,26	0,8279	1343322,52	974543,61

A.2 Caso de 100 nós

A.2.1 Algoritmo clonal

Tabela A.9: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o caso de 100 nós - Algoritmo Clonal

Solução	f_1	f_2	f_3	f_4
1cl ¹⁰⁰	6801461,28	0,6000	6931302,58	6652020,82
2cl ¹⁰⁰	6801483,55	0,6000	6930115,66	6651594,39
3cl ¹⁰⁰	6822389,15	0,2600	6954300,49	6695018,27
4cl ¹⁰⁰	6824402,73	0,2600	6956297,28	6691392,46
5cl ¹⁰⁰	6865152,77	0,2600	6996835,45	6680179,22
6cl ¹⁰⁰	7077118,72	0,6000	7196282,59	6056666,28
7cl ¹⁰⁰	7100123,67	0,6000	7219244,49	6053661,64
8cl ¹⁰⁰	7103357,89	0,6000	7222459,17	6053600,77
9cl ¹⁰⁰	7121158,09	0,6000	7240190,02	6030250,58
11cl ¹⁰⁰	7157129,70	1,0000	-	-
32cl ¹⁰⁰	7417946,10	0,0000	7543752,90	6400007,58
33cl ¹⁰⁰	7420977,83	0,0000	7546708,10	6399909,34
38cl ¹⁰⁰	7460925,59	0,1900	7584209,46	6389456,97
46cl ¹⁰⁰	7500456,10	0,1900	7621473,78	6257700,08
47cl ¹⁰⁰	7530973,29	0,1900	7651652,13	6249155,86
83cl ¹⁰⁰	7993078,35	0,9800	8152901,63	5985732,77
84cl ¹⁰⁰	7996563,72	0,9800	8156353,95	5985721,97
89cl ¹⁰⁰	8054714,52	0,9800	8173970,58	5607966,31
94cl ¹⁰⁰	8109801,11	0,6500	8229309,93	5941534,45
96cl ¹⁰⁰	8126103,24	0,5900	8241128,63	6184846,51
104cl ¹⁰⁰	8452963,96	0,9500	8612642,58	5892122,21
111cl ¹⁰⁰	8640777,61	0,9200	8768168,26	5605726,96
119cl ¹⁰⁰	8959513,13	0,5700	9073711,35	6206006,74

A.2.2 AM3

Tabela A.10: Soluções Não-Dominadas na Análise de Sensibilidade Multiobjetivo para o caso de 100 nós - AM3

Solução	f_1	f_2	f_3	f_4
1am3 ¹⁰⁰	6698459,40	1,0000	-	-
2am3 ¹⁰⁰	6698731,76	0,0000	6828316,03	6642755,41
3am3 ¹⁰⁰	6699192,38	0,0000	6825814,05	6493372,95
4am3 ¹⁰⁰	6702132,92	0,0000	6828737,81	6493364,67
5am3 ¹⁰⁰	6717052,54	0,0000	6843034,52	6423621,38
8am3 ¹⁰⁰	6739721,15	0,0000	6876730,73	6071747,97
10am3 ¹⁰⁰	6742099,90	0,0000	6869924,64	5642070,78
15am3 ¹⁰⁰	6745387,23	0,9800	6858749,73	5667555,98
19am3 ¹⁰⁰	6749497,29	0,0000	6883501,65	5454545,48
36am3 ¹⁰⁰	6772817,52	0,0000	6909785,08	5324966,74
53am3 ¹⁰⁰	6786578,27	0,0000	6926967,14	5310211,41
59am3 ¹⁰⁰	6794662,21	0,0000	6932699,26	5205106,96
86am3 ¹⁰⁰	6837425,87	0,2000	6978197,65	5120800,77
173am3 ¹⁰⁰	6938445,51	0,9800	7060824,47	4715212,44
186am3 ¹⁰⁰	6954127,44	0,0100	7088300,10	5050596,76