



UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

Eduardo Tadeu Bacalhau

Matemático – DM - UFSCAR

Métodos Heurísticos para Alocação Ótima de Recursos de Manutenção em Redes de Distribuição de Energia Elétrica

Dissertação de mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Automação.

Orientador: Prof. Dr. Christiano Lyra Filho

Co-orientador: Prof Dr. Celso Cavellucci

Banca Examinadora:

Profa. Dra. Franklina Maria Bragion de Toledo — ICMC - USP - São Carlos

Prof. Dr. Fernando José Von Zuben — FEEC - Unicamp

Campinas

Maio 2009

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

B12m Bacalhau, Eduardo Tadeu
Métodos heurísticos para alocação ótima de recursos de manutenção em redes de distribuição de energia elétrica / Eduardo Tadeu Bacalhau. --Campinas, SP: [s.n.], 2009.

Orientadores: Christiano Lyra Filho, Celso Cavellucci.

Dissertação de Mestrado - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação.

1. Confiabilidade (Engenharia). 2. Energia elétrica. 3. Heurística. 4. Otimização. 5. Simulated annealing (Matemática). I. Lyra Filho, Christiano. II. Cavellucci, Celso. III. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. IV. Título.

Título em Inglês: Heuristic methods for optimal allocation of maintenance resources on electric power distribution networks

Palavras-chave em Inglês: Reliability engineering, Electric power, Heuristic, Optimization, Simulated annealing (Matemática)

Área de concentração: Automação

Titulação: Mestre em Engenharia Elétrica

Banca examinadora: Franklina Maria Bragion de Toledo, Fernando José Von Zuben

Data da defesa: 07/05/2009

Programa de Pós Graduação: Engenharia Elétrica

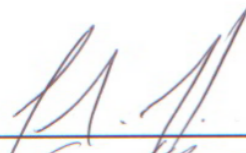
COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: Eduardo Tadeu Bacalhau

Data da Defesa: 7 de maio de 2009

Título da Tese: "Métodos Heurísticos para Alocação Ótima de Recursos de Manutenção em Redes de Distribuição de Energia Elétrica"

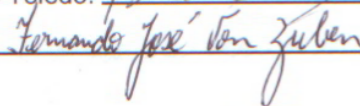
Prof. Dr. Christiano Lyra Filho (Presidente):



Profa. Dra. Franklina Maria Bragion de Toledo:



Prof. Dr. Fernando José Von Zuben:



*Ao meu pai,
João Carlos Bacalhau*

Agradecimentos

Agradeço...

Primeiramente à minha família, Mamãe, Madrinha, Padrinho, Angelita, João Carlos e João Paulo, pelo amor e carinho que recebo todos os dias.

Aos muitos amigos e irmãos de coração que participaram dessa caminhada comigo. Em especial, Roberta, Jairo, Pincher, Elisa, Ingrid, Jamile, Tahila, Marcão, Leo, Barba, Tofu, Bruna, Raquel, Natália, Jade, Thaís, Rafael (Vó), Bertola, Tiozão, Lobão, Cecília, Karina e Diego amigos e irmãos que estiveram sempre ao meu lado dando apoio.

A Iza, Edson, Jaque, Carlos, Sueli, Zézinho, Gim e Vânia, minhas diversas famílias que tenho guardadas no meu coração, que tanto me deram carinho quando estava longe de casa.

Aos amigos do Labore, Luciana, José, Daniela e em especial Fábio, pela ajuda mais do que importante na conclusão deste trabalho.

Aos amigos da república, Mormaço, Peduba e Hugo, pela amizade e apoio.

Aos dois irmãos que estão ao meu lado sempre, Laura e Cris.

Aos meus orientadores Christiano e Celso, pelas grandes sugestões e grandes ideias.

Aos professores Franklina, Fernando, Romis e Roberto, por aceitarem o convite para participar da banca examinadora dessa dissertação e pelas sugestões.

À CAPES pelo apoio financeiro.

A Deus que me deu força e sabedoria para sempre seguir em frente.

...a todos, muito obrigado.

Resumo

O trabalho aborda a alocação ótima de recursos para manutenção de redes de distribuição de energia elétrica. O modelo matemático do problema procura representar os componentes das redes, suas inter-relações para operação das redes, suas confiabilidades e mudanças de condições por ações de manutenções. O objetivo do processo de otimização é reduzir os custos de manutenção sob restrições de operação com determinados níveis de confiabilidade, ao longo de um horizonte de planejamento. As melhores estratégias de manutenção podem ser obtidas através da solução de um problema de otimização combinatória inteiro misto, com variáveis discretas e contínuas. A principal contribuição do trabalho é o desenvolvimento de meta-heurísticas híbridas que possam encontrar estratégias de manutenções de melhor qualidade, considerando-se abordagens anteriores para o problema. São desenvolvidas três meta-heurísticas híbridas, que utilizam diferentes formas de articulações de Busca em Profundidade, Busca em Largura e *Simulated Annealing*. Estudos de casos realizam comparações entre as três abordagens e um algoritmo genético híbrido. A análise dos resultados permite destacar as qualidades de cada uma das abordagens.

Palavras-chave: otimização de manutenções, manutenção baseada em confiabilidade, redes de distribuição de energia elétrica, meta-heurísticas híbridas, buscas em espaço de estados, *Simulated Annealing*.

Abstract

This work is concerned with the optimal allocation of maintenance resources on electric power distribution networks. The mathematical model represents the main components of the network, their interrelationships for the operation of the network, their reliabilities and the changes in condition due to maintenance procedures. The objective of the optimization process is the reduction of maintenance costs under constraints of reliability, during a given planning period. The best maintenance strategies can be obtained from the solution of a mixed integer combinatorial optimization problem, with continuous and discrete variables. The main contribution of the present work is the development hybrid heuristics that are able to obtain better solutions for the problem, with respect to previous approaches. Three hybrid heuristics are developed, based on the *depth first*, *breadth first* and *simulated annealing* state space searches. Case studies compare the three approaches with a hybrid genetic algorithm previously proposed to address the problem. The analysis of results highlights the benefits of each approach.

Key-words: maintenance optimization, reliability centered maintenance, electric power distribution networks, hybrid heuristics, state space searches, simulated annealing.

Lista de Figuras

2.1	Curva de Desgaste - Primeira Geração.	16
2.2	Curva da Banheira.	17
2.3	Curvas de Desgaste, adaptada de (Overman & Collard 2003).	17
2.4	Representação de redes de distribuição.	20
2.5	Rede de distribuição da Fig. 2.4 dividida em seções.	20
2.6	Representação da relação entre o modelo de taxa de falha e o modelo de otimização (Reis 2007).	22
2.7	Custo x Confiabilidade.	23
2.8	Impacto das ações de manutenção.	25
3.1	Passos do Algoritmo Genético.	29
3.2	Cadeia Binária de comprimento $L = 10$	30
3.3	Interação entre os espaços genotípico e fenotípico.	31
3.4	Representação da solução na abordagem	32
3.5	Reprodução e Mutação em cromossomos de cadeia real de comprimento $L = 10$	34
4.1	Exemplo de Grafo para a aplicação da Busca em Profundidade.	41
4.2	Grafo da Fig 4.1 na iteração seis do algoritmo de busca em profundidade.	43
4.3	Exemplo de Grafo para aplicação da Busca em Largura.	44
4.4	Grafo da Fig 4.3 na iteração seis do algoritmo de busca em largura.	46
5.1	Movimento bl_1	58
5.2	Movimento 1 - ação= bl_{1a}	58
5.3	Movimento bl_{21}	61
5.4	Movimento 2.1 - ação= bl_{21a}	61
5.5	Movimento bl_{2211}	63
5.6	Movimento 2.2.1.1 - ação= bl_{2211a}	63

5.7	Movimento bl_{2212} .	64
5.8	Movimento 2.2.1.2.	64
5.9	Movimento bl_{2213} .	65
5.10	Movimento 2.2.1.3.	65
5.11	Movimento bl_{2221} .	67
5.12	Movimento 2.2.2.1 - ação= bl_{2221a} .	68
5.13	Movimento bl_{2222} .	69
5.14	Movimento 2.2.2.2.	69
5.15	Fluxograma - Metodologia 1.	71
5.16	Fluxograma - Metodologia 2.	72
5.17	Fluxograma - Metodologia 3.	73
6.1	Rede de Referência	75
6.2	Custos Comparados - Caso I (HP=3).	81
6.3	Custos Comparados - Caso II (HP=3).	82
6.4	Boxplot 1 - Caso I - FEC = 0,55, HP=3 e Juros 15%.	84
6.5	Boxplot 2 - Caso II - FEC = 1,00, HP=3 e Juros 15%.	84
6.6	Custos Comparados - Caso I (HP=3 e HP=5).	86
6.7	Boxplot 3 - Caso I - FEC = 0,55, HP=3 e Juros 10%.	88
6.8	Boxplot 4 - Caso I - FEC = 0,55, HP=5 e Juros 10%.	88
6.9	Valores de Custos - Caso I.	90
6.10	Valores de Custos - Caso II - FEC 0,72.	91
6.11	Valores de Custos - Caso II - FEC 1,00 e FEC 1,82.	91
6.12	Boxplot 5 - Caso I - FEC = 0,55, HP=3 e Juros 15%.	92
6.13	Boxplot 6 - Caso I - FEC = 0,55, HP=5 e Juros 10%.	92
6.14	Boxplot 7 - Caso II - FEC = 1,00, HP=3 e Juros 15%.	92
6.15	Soluções AGH e BPS.	93
6.16	Solução - AGH.	94
6.17	Solução - BPS.	95

Lista de Tabelas

3.1	Correspondência entre conceitos biológicos e computacionais.	31
4.1	Correspondência entre a Interpretação da Terminologia do Sistema Físico e o Domínio Computacional.	49
6.1	Primeiro Caso de Sistema de Rede.	76
6.2	Segundo Caso de Sistema de Rede.	77
6.3	Tipos de Equipamentos.	78
6.4	Número de Clientes por Seção e Número Total de Clientes em uma Rede. . . .	78
6.5	FEC máximo e mínimo.	80
6.6	Resultados Caso Rede I.	81
6.7	Resultados Caso Rede II.	82
6.8	FEC máximo e mínimo do Caso I.	85
6.9	Resultados $HP = 3$ e $HP = 5$	86
6.10	Comparação dos Valores de Custos.	90

Sumário

1	Apresentação	13
2	Discussão e Apresentação do Modelo de Problema	15
2.1	Introdução	15
2.2	Breve Histórico	16
2.3	Manutenção Centrada na Confiabilidade - MCC	17
2.4	Apresentação do Problema	19
2.4.1	Características da Representação do Problema	21
2.4.2	Discussão do Problema	21
2.4.3	Formulação Matemática	23
2.5	Otimização Aplicada à Manutenção de Sistemas de Distribuição de Energia Elétrica	25
3	Abordagem por Algoritmo Genético	28
3.1	Introdução	28
3.2	Algoritmos Genéticos	28
3.3	AG Aplicado ao Problema de Manutenção	31
3.3.1	Representação dos Indivíduos	31
3.3.2	Procedimentos para Iniciar a População	32
3.3.3	Avaliação dos Indivíduos da População	33
3.3.4	Seleção dos Indivíduos	33
3.3.5	Recombinações e Mutações	34
3.3.6	Atualização das Populações	35
3.3.7	Critério de Parada	35
3.3.8	Parâmetros	35
3.4	Hibridização do AG para Alocação de Recursos de Manutenção	35
3.5	Análise do AG para Manutenção	36

3.5.1	Propostas de Metodologias	38
4	Estratégia de Busca no Espaço de Soluções	39
4.1	Introdução	39
4.2	Busca em Espaço de Estados	39
4.2.1	Busca em Profundidade	41
4.2.2	Busca em Largura	43
4.2.3	Hill Climbing	46
4.3	Simulated Annealing	48
4.4	Comentários	50
5	Meta-heurísticas Híbridas	52
5.1	Introdução	52
5.2	Buscas no Espaço de Estados para o Problema de Manutenção	52
5.3	A Estrutura das Novas Abordagens	53
5.3.1	Heurística Construtiva	53
5.3.2	Busca Inicial	54
5.3.3	Simulated Annealing (SA)	58
5.3.4	Fluxogramas	70
6	Estudo dos Casos	74
6.1	Introdução	74
6.2	Rede de Referência	74
6.2.1	Casos Explorados	75
6.2.2	Dados da Rede e Equipamentos	75
6.3	Experimentos	79
6.3.1	Experimento I - Variação do FEC	79
6.3.2	Análise do Experimento I	80
6.3.3	Experimento II - Horizonte de Planejamento	85
6.3.4	Análise do Experimento II	85
6.4	Estudo de Comparação das Metodologias	87
6.4.1	Análise da Comparação	89
6.5	Comentários sobre a Análise da Comparação	93
7	Conclusões	96
	Bibliografia	97

Apresentação

O trabalho aborda o problema de alocação ótima de recursos para manutenção de sistemas de distribuição de energia elétrica.

Selecionar atividades de manutenção para qualquer processo produtivo através de um método estruturado é uma das características do procedimento de manutenção centrada na confiabilidade. O objetivo principal das metodologias de manutenção aplicadas aos sistemas de distribuição de energia elétrica é obter a melhor relação possível entre custos e a confiabilidade do sistema. A otimização formal de um critério que represente o melhor uso de recursos de manutenção em sistemas de distribuição de energia elétrica é tema de estudo recente. Baseado no modelo de otimização proposto por Sittithumwat, Soudi & Tomsovic (2004), tem-se o trabalho de Reis (2007), referência que motivou os estudos aqui desenvolvidos. Seu trabalho utiliza duas abordagens por meta-heurísticas para minimizar os custos de manutenção, restritos por um nível mínimo de confiabilidade do sistema; a abordagem que obteve os melhores resultados, o Algoritmo Genético Híbrido, será objeto de comparação nos estudos de casos realizados nesta dissertação.

O problema abordado é representado por um modelo matemático que representa os componentes das redes, suas inter-relações para operação das redes, suas confiabilidades e mudanças de condições por ações de manutenções. O objetivo do processo de otimização é reduzir os custos de manutenção sob restrições de níveis mínimos de confiabilidade, ao longo de um horizonte de planejamento.

A principal contribuição do trabalho é o desenvolvimento de métodos heurísticos híbridos para encontrar as melhores estratégias de manutenções. São desenvolvidas três heurísticas, que utilizam diferentes formas de articulações entre buscas em espaços de estados, com ênfase em *busca em profundidade*, *busca em largura* e *Simulated Annealing*.

A primeira heurística desenvolvida, denominada BPS, é um procedimento construtivo guiado por uma função de avaliação do custo/benefício, seguido de uma *busca em profundidade* informada. Ao final da *busca em profundidade*, um procedimento baseado no método

Simulated Annealing é executado.

A segunda heurística desenvolvida, denominada BLS, é um procedimento construtivo guiado por uma função de avaliação do custo/benefício, seguido de *busca em largura* no espaços de estados, acrescida de informações sobre o problema para reduzir o espaço de busca. Ao final da *busca em largura*, um procedimento baseado no método *Simulated Annealing* é executado.

A terceira heurística desenvolvida, denominada BSA, é um procedimento construtivo guiado por uma função de avaliação do custo/benefício, seguido apenas de uma versão do método *Simulated Annealing* adaptada às características do problema.

Serão realizados dois conjuntos de estudos de casos. O primeiro conjunto irá comparar os procedimentos de otimização desenvolvidos entre si, enquanto que o segundo conjunto irá comparar os métodos desenvolvidos com o algoritmo genético híbrido proposto por Reis (2007). A análise dos resultados permite destacar as qualidades de cada uma das abordagens.

O próximo capítulo (Capítulo 2) apresenta um breve histórico sobre estratégias de manutenções baseadas em confiabilidade, uma revisão bibliográfica e a formulação do problema de otimização para encontrar as melhores estratégias de manutenções. O Capítulo 3 apresenta os principais conceitos de algoritmos genéticos, discute o algoritmo genético híbrido proposto por Reis (2007) e apresenta considerações para aperfeiçoamento da estratégia de solução do problema. No Capítulo 4 são discutidos conceitos de buscas em espaços de estados, utilizados para o desenvolvimento dos algoritmos propostos neste trabalho. O Capítulo 5 discute as meta-heurísticas híbridas baseados em buscas em espaço de estados, desenvolvidas para abordagem do problema. O Capítulo 6 é dedicado aos estudos de casos e análise dos resultados. As conclusões sobre o trabalho são apresentadas no Capítulo 7.

Discussão e Apresentação do Modelo de Problema

2.1 Introdução

A metodologia de Manutenção Centrada na Confiabilidade (MCC) surgiu na década de 60, através de uma força-tarefa iniciada por uma grande fábrica de aeronaves americana (Siqueira 2005).

Foi desenvolvida e aprimorada a partir da experiência adquirida ao longo dos anos pela necessidade de adaptação aos meios em que estava sendo aplicada. Atualmente, é reconhecida e aplicada em diversos setores, sempre com a finalidade de encontrar um modelo de implementação consistente para cada empreendimento.

O envelhecimento dos sistemas de distribuição de energia nos E.U.A. e na Europa motivou o estudo da aplicação da metodologia de manutenção nos sistemas de distribuição de energia elétrica (Billinton & Billinton 1989), especialmente para determinar o impacto do desgaste da rede na confiabilidade do sistema (Brown & Willis 2006, Dominelli, Rao & Kundur 2006, Endrenyi & Anders 2006, Li & Guo 2006, Li, Vaahedi & Choudhury 2006).

Este capítulo apresenta um breve resumo do desenvolvimento dos conceitos que apoiam decisões sobre manutenções. Em seguida, discutem um modelo matemático para otimização de políticas de manutenção em sistemas de distribuição de energia elétrica, tendo como referência a representação proposta por Reis (2007). A seção seguinte do capítulo apresenta a formulação matemática de problema de otimização da manutenção de redes de distribuição de energia elétrica, que também será discutida ao longo dos próximos capítulos. Por fim, discute bibliografia sobre técnicas de otimização aplicadas a problemas de manutenção em sistemas de distribuição de energia elétrica, com ênfase no trabalho de Reis (2007).

2.2 Breve Histórico

O evento que marca a origem da MCC é a criação de uma força-tarefa Americana que buscava atender as exigências de qualidade das autoridades aeronáuticas para a certificação do aumento no número de assentos proposto em relação à aeronave anterior. Depois desse primeiro trabalho, seguiram-se duas novas Forças-Tarefa que buscavam também melhorias em alguns modelos de aeronaves. Logo em seguida, o governo americano adotou a metodologia desenvolvida na marinha, ampliando depois para outras áreas militares e civis (Siqueira 2005). A partir do terceiro relatório da força-tarefa, várias versões da metodologia foram elaboradas buscando adaptações para novas aplicações.

Moubray (1997) classifica a história moderna dos procedimentos de manutenção em três gerações: Mecanização, Industrialização e Automatização.

- Primeira Geração - Mecanização:

Seguindo a cronologia histórica, a primeira geração surgiu do crescimento da mecanização da indústria pós-guerra, onde a preocupação era sobretudo com a quebra do equipamento, ou seja, os investimentos eram destinados à substituição do equipamento, pois a prevenção de falha não era prioridade. A Fig. 2.1 mostra uma interpretação da curva de desgaste de um equipamento sob o ponto de vista da primeira geração.

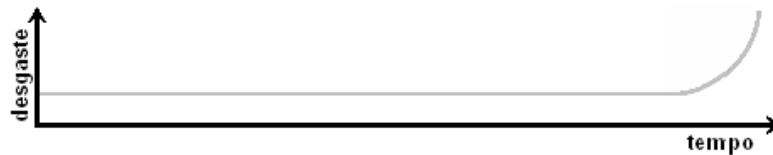


Figura 2.1: Curva de Desgaste - Primeira Geração.

- Segunda Geração - Industrialização:

Devido à automatização do processo de produção industrial, houve uma mudança no pensamento em relação à longevidade do equipamento e ao controle dos recursos previstos para os investimentos.

Com o planejamento e o controle do sistema de produção, além da redução dos custos disponíveis, tem-se o começo da aplicação da ideia de manutenção preventiva. Esse conceito traz uma nova representação para a curva de desgaste de um equipamento. Essa representação é conhecida como a ‘curva da banheira’ (Moubray 1997), que é ilustrada na Fig. 2.2.

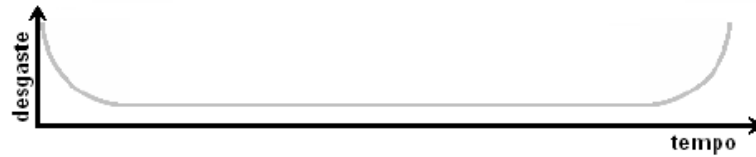


Figura 2.2: Curva da Banheira.

- Terceira Geração - Automatização:

O crescimento do consumo junto com o aumento da expectativa da população em relação a qualidade de serviços foram as causas das inovações em procedimentos de manutenção, que levaram ao surgimento da MCC. A principal inovação foi a implantação do conceito de confiabilidade do sistema, motivada por novos objetivos, tais como a busca pela eficácia da manutenção, as melhorias na qualidade dos produtos, a preocupação com a extensão de vida dos equipamentos e percepção da necessidade de preservação ambiental. Esses objetivos exigiram novas técnicas de manutenção, que possibilitem um melhor monitoramento dos equipamentos. A curva de desgaste não é mais única, mas apresenta uma característica própria para cada tipo de equipamento, como ilustra a Fig. 2.3 (Overman & Collard 2003).

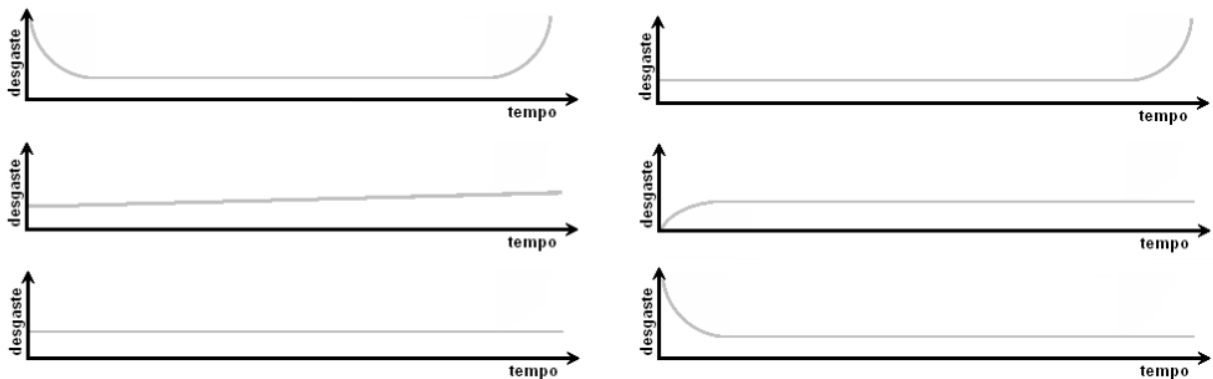


Figura 2.3: Curvas de Desgaste, adaptada de (Overman & Collard 2003).

A utilidade da MCC foi percebida e estendida para outras áreas, tais como o setor elétrico, foco deste trabalho.

2.3 Manutenção Centrada na Confiabilidade - MCC

Selecionar atividades de manutenção para qualquer processo produtivo através de um método estruturado é uma das características da manutenção centrada na confiabilidade. As

atividades podem ser classificadas de acordo com a forma de programação e o objetivo das tarefas a serem executadas. Para se definir os objetivos da MCC, a padronização (normalização) da metodologia foi essencial. Segundo a norma, cabe à MCC ‘*determinar requisitos de manutenção para modos de falha que possam causar falhas funcionais de quaisquer itens físicos em seu ambiente operacional*’ (Moubray 1997). Dada a definição, a MCC pode ser apresentada com os seguintes objetivos (Siqueira 2005):

- Preservar as funções dos equipamentos, com a segurança requerida;
- Restaurar sua confiabilidade e segurança projetada, após a deterioração;
- Otimizar a disponibilidade;
- Minimizar o custo do ciclo de vida;
- Atuar conforme os modos de falha;
- Realizar apenas as atividades que precisam ser feitas;
- Documentar as razões para escolha das atividades; e
- Agir em função dos efeitos e reduzir as consequências das falhas.

Dentre esses objetivos, o principal é reduzir as consequências das falhas. Assim, são elaboradas sete questões que levam à estruturação de uma sequência de implementação (Siqueira 2005, Moubray 1997).

1. Quais são as funções devem ser preservadas?
2. Quais são as falhas funcionais?
3. Quais são os modos de falha?
4. Quais são os efeitos das falhas?
5. Quais são as consequências das falhas?
6. Quais são as tarefas aplicáveis e efetivas?
7. Quais são as alternativas restantes?

Dadas as questões formuladas acima, a metodologia busca respostas através de uma sequência de ações, que caracteriza a implantação da MCC:

1. *Seleção de Sistema e Coleta de Informações:*
Identifica e documenta o sistema ou processo que será submetido à análise;
2. *Análise de Modos de Falhas e de Efeitos:*
Identifica e documenta os efeitos de todas as falhas e seus modos de falha;
3. *Seleção de Funções Significantes:*
Analisa cada função identificada através de um processo estruturado;
4. *Seleção de Atividades Aplicáveis:*
Determina-se as tarefas de manutenção preventiva aplicáveis;
5. *Avaliação da Efetividade das Atividades:*
Analisa a tarefa de manutenção preventiva através de um processo estruturado;
6. *Seleção das Atividades Aplicáveis e Efetivas:*
Analisa qual é a melhor tarefa através de um outro processo estruturado; e
7. *Definição da Periodicidade das Atividades:*
Métodos e critérios são estabelecidos para definição da frequência de execução das atividades.

Na próxima seção é apresentado o modelo de problema desenvolvido neste trabalho para a elaboração dos estudos de casos.

2.4 Apresentação do Problema

Em um sistema de distribuição de energia elétrica (SDEE), cada equipamento possui uma função específica. Têm-se equipamentos de manobras, de proteção, de correção e de transformação, condutores e infra-estruturas associadas, como postes, cruzetas e amarrações, que partem das subestações e transportam a energia até os consumidores (Reis 2007).

Os equipamentos de proteção (disjuntores, chaves-fusíveis, religadores e chaves-faca) protegem a rede da propagação de falhas. Esses equipamentos podem ser utilizados para caracterizar as fronteiras das seções, como sugerem Sittithumwat et al. (2004) e Reis (2007).

Em redes de distribuição radiais, se qualquer equipamento de uma seção falhar, o equipamento de proteção é acionado, causando a interrupção na transmissão da energia. Isto implica que as seções mais próximas dos alimentadores possuem importância maior em relação à confiabilidade do sistema, pois se ocorrerem falhas um número maior de consumidores será atingido.

A Fig. 2.4 representa uma rede de distribuição simplificada (Reis 2007), com dois alimentadores e quatro equipamentos de proteção. A rede está dividida nas Seções A, B, C, D e

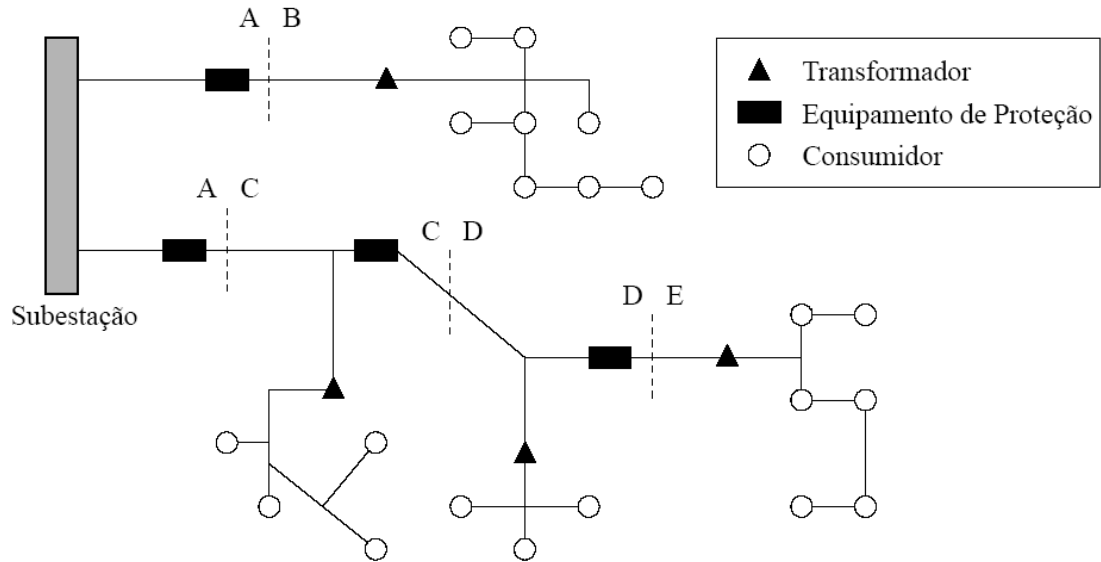


Figura 2.4: Representação de redes de distribuição.

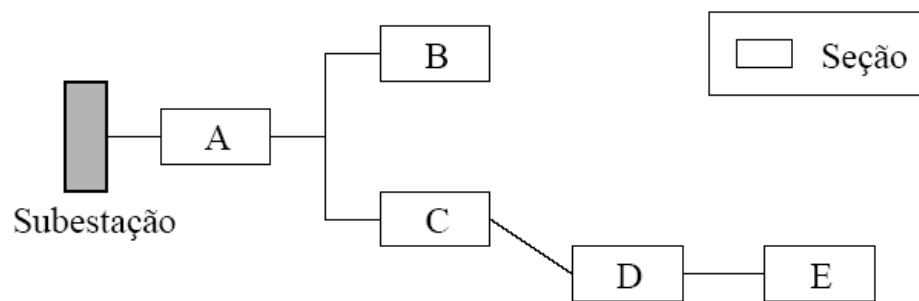


Figura 2.5: Rede de distribuição da Fig. 2.4 dividida em seções.

E, delimitadas pelas linhas pontilhadas. A Fig 2.5 ilustra a rede com a separação por seções (Reis 2007).

2.4.1 Características da Representação do Problema

A representação do sistema de distribuição por blocos (seções) foi estudada por Sittithumwat et al. (2004) e Reis (2007); ela permite avaliar o impacto das ações de manutenção na confiabilidade de cada bloco e da rede como um todo, levando em consideração o número de clientes de cada bloco. Logo, a representação é adequada para encontrar políticas de manutenção que buscam minimizar os custos de manutenção sob restrições de confiabilidade.

O modelo proposto por Reis (2007) procura estimar o processo de deteriorização dos equipamentos ao longo dos anos e suas mudanças de estados por ações de manutenção. A Fig. 2.6 ilustra a inter-relação entre os dois pontos descritos na representação. A partir dessas informações e do impacto das falhas na confiabilidade das redes, é possível desenvolver metodologias para encontrar as melhores alternativas de manutenção.

2.4.2 Discussão do Problema

Pode-se encontrar diferentes alternativas de construção da função objetivo. Uma delas busca juntar os custos de manutenções decorrentes de baixa confiabilidade (corretiva) com os custos de manutenções programadas (preventiva). Os custos de manutenções decorrentes de baixa confiabilidade (CBC) e os custos de manutenções programadas (CMP) são ilustrados na Fig. 2.7 em função da confiabilidade. Uma boa estratégia de otimização dos recursos é encontrar o ponto de mínimo da curva de custo total, que corresponde à soma dos custos CBC e CMP.

Adotando-se como função objetivo (a ser minimizada) a soma dos custos de manutenções corretivas e preventivas, o índice de confiabilidade mínima do sistema é considerado como restrição. Para avaliação da confiabilidade, pode-se adotar a *frequência equivalente de interrupções por unidade consumidora* (FEC). As ações de manutenção são as variáveis de decisão, ou seja, os níveis de manutenção de cada equipamento determinam os valores de custo total e as taxas de falha dos equipamentos, que estão diretamente relacionadas com a confiabilidade do sistema. O número de níveis de manutenção depende do tipo do sistema analisado, mas geralmente o maior número considerado é de três níveis de manutenção (IEEE Standard 493 1998).

É importante ressaltar que outras possibilidades de função objetivo para a representação do problema podem ser adotadas. Por exemplo, pode-se tentar maximizar a confiabilidade do sistema, como em (Sittithumwat et al. 2004), que procura minimizar o índice FEC sujeito à disponibilidade de recursos para a manutenção em um determinado período. Pode-se também

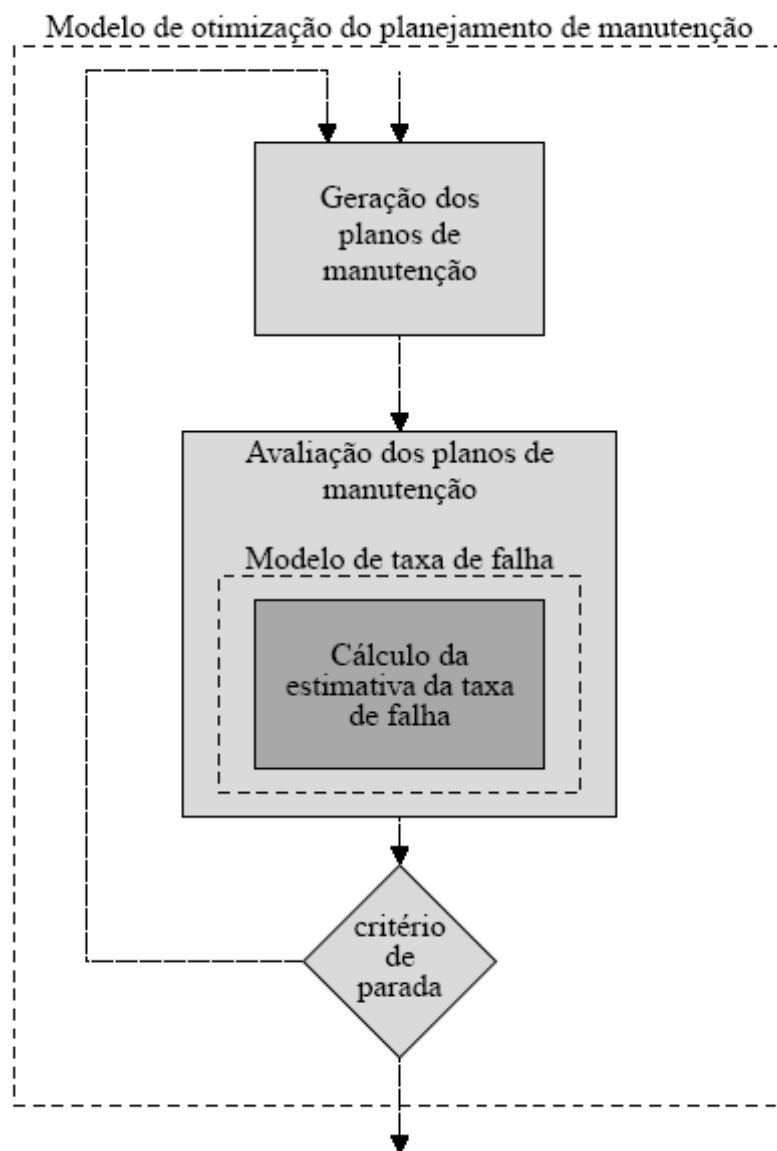


Figura 2.6: Representação da relação entre o modelo de taxa de falha e o modelo de otimização (Reis 2007).

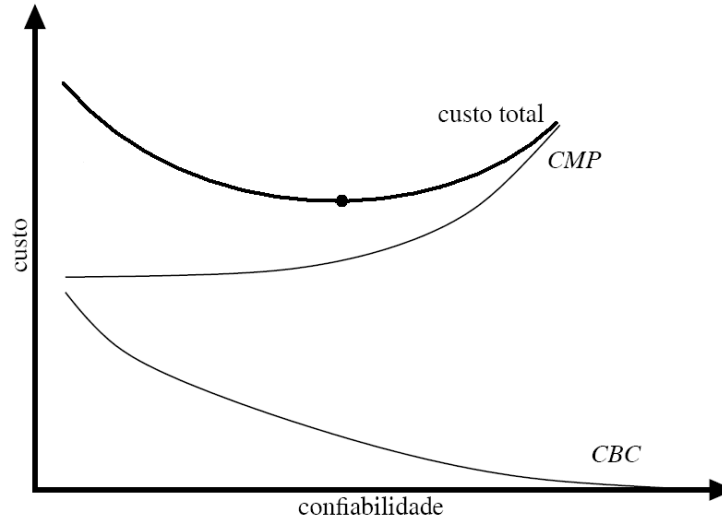


Figura 2.7: Custo x Confiabilidade.

procurar minimizar o custo de manutenção e maximizar a confiabilidade do sistema, através de uma abordagem multiobjetivo.

Apresentadas as características da representação, pode-se formalizar matematicamente um problema de otimização com este modelo para buscar as melhores políticas de manutenções.

2.4.3 Formulação Matemática

O modelo a ser adotado foi desenvolvido por Reis (2007), tendo como referência o trabalho de Sittithumwat et al. (2004). A função objetivo do problema busca minimizar o custo de manutenção dos equipamentos, tendo como restrições as metas de confiabilidade do sistema.

A confiabilidade do sistema é medida pelo índice *FEC* (*Frequência Equivalente de Interrupções por Unidade Consumidora*), que indica a média de interrupções do sistema em um determinado período. O *FEC* é calculado a partir das taxas de falhas das seções e do número de clientes da seção, em um período. É dado por:

$$FEC^t = \frac{\sum_{s \in S} \lambda_s^t N_s}{N_T}, \quad (2.1)$$

onde:

- S é o conjunto de todas as seções;
- λ_s^t é a taxa de falha da Seção s no período t ;
- N_s é o número de clientes na Seção s ;
- N_T é o número total de clientes da rede.

Para representar a confiabilidade de cada equipamento, é necessário calcular a taxa de falha individualmente, incluindo suas modificações por ações de manutenção. As taxas de falhas dos equipamentos e das seções no período t podem ser calculadas pelas seguintes equações:

$$\lambda_e^t = \lambda_e^{(t-1)} \sum_{n \in N_{k_e}} \delta_{k_e n} x_{en}^t, \quad (2.2)$$

$$\sum_{n \in N_{k_e}} x_{en}^t = 1 \quad (2.3)$$

$$\lambda_s^t = \lambda_s + \sum_{e \in E_s} \lambda_e^t, \quad (2.4)$$

$$(2.5)$$

onde:

- λ_e^t é a taxa de falha do equipamento e no período t ;
- $\lambda_e^{(t-1)}$ a taxa de falha do equipamento e no ano anterior ou a taxa de falha inicial do equipamento $e(t = 1)$;
- N_{k_e} o conjunto de ações preventivas que podem ser encontradas;
- $\delta_{k_e n}$ o multiplicador de taxa de falha do equipamento de tipo k_e para a ação de nível n ;
- x_{en}^t a variável booleana de decisão que indica se o equipamento e sofreu ($x_{en}^t = 1$) ou não ($x_{en}^t = 0$) uma manutenção de nível n no período t .

Na Equação 2.4, λ_s é a taxa de falha base da Seção s , que representa a parcela da taxa de falha associada a fatores não incorporados no modelo (é constante). Têm-se N_{k_e} níveis de manutenção para cada equipamento, isso implica que existirá um $\delta_{k_e n}$ diferente para cada ação, o que determina diferentes formas de atualização da taxa de falha do equipamento.

Usando-se a representação de rede descrita, pode-se caracterizar o problema de otimização a ser resolvido para encontrar as melhores políticas de manutenção em sistemas de distribuição de energia elétrica, como dado a seguir:

$$\min_{\forall x_{en}^t} \sum_{t=1}^{HP} \left\{ \sum_{e \in E} \left[\sum_{n \in N_{k_e}} (p_{k_e n} x_{en}^t) + \lambda_e^t C_{k_e} \right] \times \alpha_t \right\} \quad (2.6)$$

$$s.a : \quad FEC^t \leq FEC_{perm} \quad \forall t = 1, \dots, HP, \quad (2.7)$$

onde:

- E é o conjunto que contém todos os equipamentos sujeitos à manutenção;
- FEC_{perm} é o valor máximo permitido para o FEC;
- $p_{k_e n}$ é o custo da ação preventiva de nível n no equipamento de tipo k_e ;
- c_{k_e} é o custo da manutenção corretiva do equipamento de tipo k_e ;
- α_t é um parâmetro associado a cada intervalo de tempo.

O parâmetro α_t visa equilibrar a distribuição das ações de manutenção ao longo dos intervalos do horizonte de planejamento (HP). Nessa formulação, α_t pode ser definido como o ‘valor presente dos investimentos’:

$$\alpha_t = \left(\frac{1}{1+j} \right)^t, \quad (2.8)$$

onde t é o período e j a taxa de juros aplicada ao investimento.

As variáveis de decisão (ações de manutenção em cada equipamento) determinam os valores do custo total de manutenção e os multiplicadores de taxa de falha de cada equipamento. Os multiplicadores definem os valores para atualização da taxa de falha de um equipamento ao longo dos períodos, representados na Equação 2.2.

O impacto das ações de manutenção nos custos e na qualidade do serviço de distribuição (FEC) são ilustradas na Fig.2.8.



Figura 2.8: Impacto das ações de manutenção.

A próxima seção discute trabalhos sobre aplicação de procedimentos de manutenção baseados em confiabilidade aliados a técnicas matemáticas de otimização.

2.5 Otimização Aplicada à Manutenção de Sistemas de Distribuição de Energia Elétrica

O objetivo principal das metodologias de manutenção aplicadas aos sistemas de distribuição de energia elétrica é maximizar a confiabilidade do sistema. Alocando-se os recursos da melhor maneira possível, pode-se obter benefícios quanto a confiabilidade dos sistemas e reduzir os custos operacionais (Cavellucci, González & Lyra 2006).

Neste contexto, encontram-se na literatura vários métodos que têm como base a otimização da manutenção. Dentre esses métodos vê-se em Hilber & Bertling (2008), Bertling, Miranda, Matos & Hilber (2007) e Hilber (2008) aplicações de otimização multiobjetivo, visando melhorar a confiabilidade e reduzir os custos de manutenção da rede. O objetivo é encontrar várias soluções eficientes, representando os melhores compromissos entre os custos e desempenhos da rede. Bertling et al. (2007) desenvolveu uma meta-heurística, que utiliza o algoritmo de otimização *Particle Swarm* (Nuvem de Partículas), com a finalidade de aplicá-la ao sistema de distribuição urbana de Estocolmo, Suécia (Bertling et al. 2007, Hilber 2008). A metodologia foi também aplicada ao sistema de distribuição da área urbana da cidade de Kristinehamn em conjunto com a área rural localizada nos arredores da cidade (Hilber & Bertling 2008). A metodologia propõe utilizar o método de Simulação de Monte Carlo para calcular o índice de importância associada à confiabilidade de cada equipamento, identificando as mudanças dos estados desses equipamentos. Obtêm-se assim dados de confiabilidade do sistema, tais como o custo de interrupção, o FEC (Frequência Equivalente de Interrupções por Unidade Consumidora) e o DEC (Duração Equivalente de Interrupção por Unidade Consumidora), possibilitando uma análise de sensibilidade do desempenho da rede. Esses trabalhos usam uma metodologia de otimização multiobjetivo para encontrar os melhores compromissos entre custos e confiabilidade.

Usando uma metodologia baseada no modelo probabilístico AMP (*Asset Management Planner*), Endrenyi, Anders & Leite da Silva (1998) procuram descrever o impacto do processo de deteriorização dos elementos na confiabilidade do sistema. O trabalho usa um método Markoviano para avaliar o impacto das variáveis de decisão na confiabilidade do sistema. Fazem também um estudo de sensibilidade nas soluções, a fim de determinar os melhores valores para as variáveis de decisão.

O modelo de otimização proposto por Sittithumwat et al. (2004) tem como principal objetivo maximizar a confiabilidade do sistema baseado nas informações fornecidas pelos índices da confiabilidade dos equipamentos, considerando restrições de investimento e mão-de-obra disponível. A metodologia proposta foi aplicada a um sistema radial simples, que representa um típico alimentador principal com diversos ramos laterais, considerando três níveis de manutenção (intensiva, mínima e nenhuma manutenção). As melhores políticas de manutenção são encontradas através da solução de um problema de programação linear inteiro-mista. Projetando cenários com informações limitadas, o artigo utiliza a programação linear com variáveis nebulosas. Nos estudos de casos, três cenários para otimização são analisados: o primeiro cenário determina o nível de manutenção usando programação linear binária para casos com informações completas; o segundo cenário, determina o nível de manutenção usando programação linear nebulosa para casos com informações limitadas; o terceiro investiga o custo de adição de informações para a rede.

Kuntz, Christie & Venkata (2002) apresentam um modelo para planejamento da manu-

tenção de uma determinada vegetação. Esse modelo tem como objetivo determinar quando e onde aplicar a manutenção (cortes), sujeito a restrições de confiabilidade, custo e mão-de-obra disponível. O artigo usa um modelo de otimização combinatória, solucionado por técnicas heurísticas de busca. O critério de confiabilidade usa uma ponderação dos dois índices de continuidade no fornecimento: a frequência equivalente de interrupções por unidade consumidora (FEC) e a duração média de interrupções por cliente (DEC). Uma rede neural procura estabelecer a ligação entre o planejamento de manutenção e a confiabilidade do sistema. Três métodos heurísticos de busca são explorados para abordar o problema de planejamento de manutenção ótimo: algoritmo genético, *simulated annealing* e busca tabu. A técnica de busca local do tipo *hill-climbing* foi também explorada, associada a cada uma das três técnicas. O algoritmo genético combinado com a técnica de *hill-climbing* apresentou os melhores resultados ao final do processamento.

Considerando a gestão de riscos, Janjic & Popovic (2007) propuseram uma abordagem com o objetivo de determinar o plano ótimo de ações ao longo dos anos. O modelo de transição de estados se baseou em uma árvore de decisões que representa os possíveis cenários anuais de manutenções preventivas. Um método de programação dinâmica foi utilizado para minimizar o custo total esperado de manutenção preventiva, restrito a um limite máximo para o custo anual de manutenção.

Baseado no modelo de otimização proposto por Sittithumwat et al. (2004), o trabalho de Reis (2007) utiliza duas abordagens de meta-heurísticas para minimizar os custos de manutenção, restritos por um nível mínimo de confiabilidade do sistema. A primeira abordagem foi baseada no método GRASP (*greedy randomized adaptive search procedure*) (Feo & Resende 1995). GRASP é um procedimento iterativo com múltiplos reinícios, composto de duas fases: construção e busca local (Cavellucci, França, Lyra & Reis 2007). As primeiras soluções são geradas por uma heurística construtiva que usa uma relação custo/benefício para aumentar, ou não, o nível de manutenção de um dado elemento. A busca local procura usar conhecimentos adicionais sobre o problema para melhorar as soluções obtidas na etapa de construção. A segunda abordagem desenvolvida por Reis (2007) é baseada em um algoritmo genético híbrido (AGH). Essa abordagem é detalhada no próximo capítulo. A aplicação das abordagens em redes reais de grande porte destacam o algoritmo genético híbrido, que obteve melhores resultados, tanto na qualidade das soluções quanto no tempo de execução.

O próximo capítulo faz um resumo das ideias de algoritmos genéticos, seguido da apresentação do algoritmo genético híbrido, desenvolvido por Reis (2007).

Abordagem por Algoritmo Genético

3.1 Introdução

Os algoritmos genéticos são uma classe particular de algoritmos populacionais que usam técnicas inspiradas pela biologia evolutiva.

Este capítulo discute os principais conceitos de algoritmos genéticos e, em particular, o algoritmo genético híbrido desenvolvido por Reis (2007) para otimização de planos de manutenção de redes de distribuição de energia elétrica. Ao final do capítulo, são discutidos aspectos que podem ser aperfeiçoados na metodologia desenvolvida por Reis (2007).

As ideias apresentadas no capítulo servirão como referência para o desenvolvimento de uma nova metodologia para abordagem do problema de otimização de estratégias de manutenção de redes de distribuição de energia elétrica, apresentada no próximo capítulo.

3.2 Algoritmos Genéticos

Desenvolvido por Holland no livro *Adaptation in Natural and Artificial Systems* (1975), um algoritmo genético (AG) procura codificar computacionalmente os fenômenos de reprodução, evolução e adaptação, com a finalidade de encontrar soluções para problemas de otimização.

A estratégia dos algoritmos genéticos é tratar as possíveis soluções de um problema como *indivíduos* de uma *população* que, a cada iteração, procura *evoluir*, ou seja, obter soluções melhores. Esse processo pode ser traduzido em um modelo de otimização, resumido nos seguintes passos:

- Cria uma população inicial;
- Avalia a população de indivíduos através de uma função de aptidão;
- Seleciona os melhores indivíduos e determinar uma nova população;

- Realiza operações de recombinação e mutação dos indivíduos, levando em conta seus valores de aptidão;
- Determina o fim do algoritmo através de um critério de parada.

Esses passos são ilustrados na Fig.3.1 e detalhados no Pseudo-código 1.

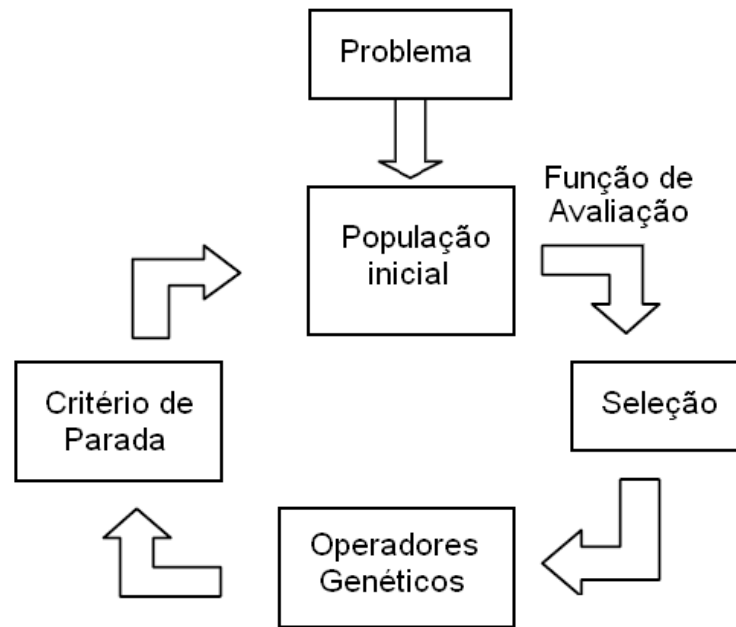


Figura 3.1: Passos do Algoritmo Genético.

Pseudo-código 1 Algoritmo Genético

```

1:  $t \leftarrow 0$ 
2: inicializar  $P(t)$ 
3: avaliar  $P(t)$ 
4: enquanto não atingir a condição de parada faça
5:    $t \leftarrow t + 1$ 
6:   selecionar  $P(t)$  de  $P(t - 1)$ 
7:   realizar cruzamentos entre os indivíduos selecionados de  $P(t)$ 
8:   realizar mutações em indivíduos de  $P(t)$ 
9:   avaliar  $P(t)$ 
10: fim enquanto
  
```

O AG representado pelo Pseudo-código 1 mantém uma população de indivíduos $P(t)$ para cada iteração t do algoritmo. A população inicial de indivíduos, $P(0)$, é iniciada na Etapa 2 e cada indivíduo é avaliado pela função *fitness* na Etapa 3. Na Etapa 6 os indivíduos mais adaptados (aqueles com melhores *fitness*) têm a maior probabilidade de serem selecionados para gerar descendentes, novas soluções potenciais do problema (Etapa 7). Após os descendentes terem sido gerados pelo operador genético de reprodução na Etapa 7, o operador de

mutação é aplicado a esses descendentes (Etapa 8). A nova população é avaliada na Etapa 9. As etapas 5 a 9 são executadas repetidamente até que a condição de parada seja satisfeita.

Os indivíduos são definidos em dois espaços:

- Espaço Genotípico

É o espaço que contém a população de indivíduos caracterizados por uma codificação, cromossomos, que representam uma solução potencial do problema. Os cromossomos podem ser apresentados por uma cadeia de atributos associados à solução do problema a ser resolvido, chamados genes. Cada gene tem uma localização definida (*locus*) e um valor associado chamado alelo (Castro 2006). A Fig. 3.2 exemplifica a estrutura de um cromossomo representada por uma cadeia de atributos, onde cada atributo corresponde aqui a um dígito binário $\{0, 1\}$. Um exemplo de representação de uma cadeia de atributos com dígitos reais é ilustrada na abordagem desenvolvida por Reis (2007), apresentada na próxima seção.

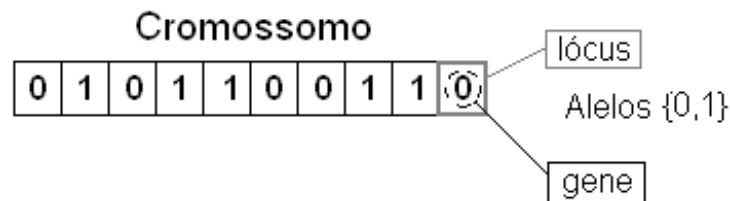


Figura 3.2: Cadeia Binária de comprimento $L = 10$

- Espaço Fenotípico

É o espaço que contém a população de indivíduos caracterizados pela manifestação do genótipo no comportamento do indivíduo, como um produto de sua interação com o ambiente. Essa manifestação é definida por uma função de adaptação. Para representar uma medida desse grau de aptidão dos indivíduos, tem-se a função *fitness*. A função *fitness* define um critério para a seleção daqueles que participarão do processo de recombinação.

A interação desses dois espaços no algoritmo genético é ilustrada pela Fig.3.3(Atmar 1994).

Na Fig.3.3, a passagem $g_1 \leftarrow f_1$ representa a decodificação da população inicial definida no espaço genotípico para o espaço fenotípico, $f_1 \leftarrow f_2$ representa o processo de seleção através da função de aptidão dos elementos da população inicial no espaço fenotípico, $f_2 \leftarrow g_1$ representa codificação da população selecionada pela função de aptidão do problema no espaço fenotípico para o espaço genotípico, $g_2 \leftarrow g'_1$ representa a população selecionada passando pelo processo de recombinação e mutação no espaço genotípico, representando a nova geração da população produzida pelo algoritmo g'_1 .

A Tabela 3.1 (Castro 2006) (Goldberg 1989) resume algumas associações entre conceitos de biologia usados em algoritmos genéticos.

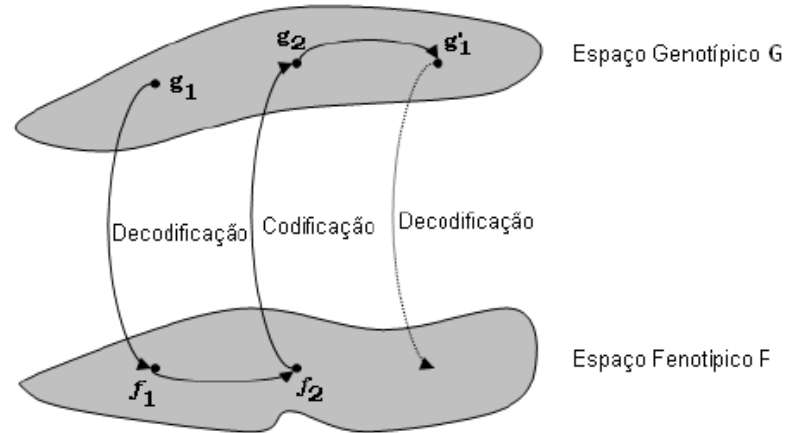


Figura 3.3: Interação entre os espaços genotípico e fenotípico.

Biologia	Algoritmos Genéticos
Cromossomo	Estrutura de dados associada às soluções potenciais do problema
Gene	Elemento ocupando uma dada posição na estrutura de dados
Locus	Posição ocupada por um gene na estrutura de dados
Alelo	Diferentes valores que podem ser atribuídos ao gene
Reprodução	Combinação de partes da estrutura de dados para gerar novas soluções
Mutação	Modificação do valor de um ou mais genes do cromossomo
Fitness	Valor que indica a qualidade de um indivíduo como solução de um problema
Seleção	Operador que realiza a seleção dos melhores indivíduos para a seleção
Genótipo	Codificação que representa uma solução potencial do problema
Fenótipo	Expressão do genótipo associado às características do problema em estudo

Tabela 3.1: Correspondência entre conceitos biológicos e computacionais.

3.3 AG Aplicado ao Problema de Manutenção

Algoritmos genéticos podem ser aplicados para resolver o problema de manutenção de sistemas de distribuição de energia elétrica, formulado na Seção 2.4.3. A abordagem desenvolvida por Reis (2007) é resumida nos itens a seguir.

3.3.1 Representação dos Indivíduos

A escolha da representação de um indivíduo junto com os operadores utilizados para alterar seu conteúdo são os principais componentes em um algoritmo evolutivo (Michalewicz 1996).

Na metodologia desenvolvida por Reis (2007) os indivíduos são representados como um

vetor de números reais (Fig. 3.4).

$$\begin{array}{|c|c|c|c|} \hline inv_{e_1} & inv_{e_2} & \dots & inv_{e_n} \\ \hline e_1 & e_2 & & e_n \\ \hline \end{array} \quad inv_{e_i} \in [0, 1]$$

Figura 3.4: Representação da solução na abordagem

Como ilustrado na Fig. 3.4, cada elemento da rede de distribuição possui uma posição (*locus*) neste vetor, indicando o valor do investimento destinado à manutenção desse elemento. Os valores de investimentos serão normalizados, ou seja, os alelos estão definidos no intervalo entre 0 e 1.

Tendo em vista as restrições do problema, a abordagem utiliza um processo de decodificação¹ que prevê o tratamento de indivíduos inactiváveis.

A representação dos indivíduos por um vetor de números reais não representa diretamente uma solução. Ela define quanto recurso destinar para cada elemento, sem determinar quais ações de manutenção devem ser programadas.

Determinada a representação de cada indivíduo, a prioridade de cada elemento (P_e) é calculada pela seguinte equação:

$$P_e = inv_e - invJaAlocado_e, \quad (3.1)$$

sendo inv_e o investimento destinado ao elemento e e $invJaAlocado_e$ o investimento já determinado para o elemento e .

De acordo com a prioridade calculada, pode-se selecionar, a cada iteração, as ações de manutenção programadas para os investimento alocados.

3.3.2 Procedimentos para Iniciar a População

Gerar aleatoriamente uma solução é uma alternativa simples de inicializar os indivíduos da população em um AG. Contudo, o uso de informações adicionais sobre cada elemento da rede pode levar a um ganho de tempo na resolução. Essa estratégia de geração de indivíduos é boa desde que se evite a baixa diversidade da população e, conseqüentemente, pontos de convergência prematura (ótimos locais).

Na abordagem proposta por Reis (2007) é desenvolvida uma heurística construtiva associada a uma função de avaliação que determina a relação custo/benefício, visando aumentar, ou diminuir, o nível de manutenção nos elementos da rede.

¹A ilustração do processo de decodificação, assim como o Pseudo-código referente, estão definidos e exemplificados em (Reis 2007).

A função de avaliação para o problema de alocação de recursos na manutenção é representada da seguinte forma:

$$FG(e) = \frac{custo_{e,maximo} - custo_{e,atual}}{contFEC_{e,atual} - contFEC_{e,minimo}}, \quad (3.2)$$

sendo $contFEC_{e,atual}$ a contribuição atual do elemento e para o FEC total do sistema, $contFEC_{e,minimo}$ o valor da contribuição desse elemento para o FEC no maior nível de confiabilidade, $custo_{e,maximo}$ o custo do maior nível de manutenção e $custo_{e,atual}$ o custo do nível atual de manutenção.

A heurística construtiva para inicializar a população consiste em selecionar um elemento na população e escolher seu nível de manutenção (ação programada) utilizando a Equação 3.2. Esse processo se repete até que a população atinja a factibilidade. O objetivo da heurística construtiva é gerar boas soluções iniciais para o AG, buscando sempre a diversidade na população para uma boa exploração do espaço de busca.

3.3.3 Avaliação dos Indivíduos da População

Os indivíduos da população são avaliados por meio da função de avaliação associada a função objetivo do problema de alocação ótima de recursos de manutenção. A formulação desta função é representada pela Equação 3.3.

O algoritmo genético deve buscar soluções que representem valores mínimos para a Equação 2.6. Porém, como normalmente os AG's visam a maximização da função de aptidão (*fitness*) (Holland 1975) dos indivíduos, a função de aptidão pode ser considerada como:

$$\frac{1}{F(soluc\ao)}, \quad (3.3)$$

sendo $F(soluc\ao)$ o valor da Equação 2.6.

3.3.4 Seleção dos Indivíduos

A seleção determina a próxima geração baseada na probabilidade de cada indivíduo ser escolhido. Existem na literatura várias formas dessa seleção acontecer (Bäck, Fogel & Michalewicz 2000). Reis (2007) usa a seleção por torneio, onde um conjunto de indivíduos é escolhido aleatoriamente na população. Em seguida, o conjunto de indivíduos escolhidos participa de um torneio que seleciona deterministicamente o melhor indivíduo, dependendo do seu *fitness*.

3.3.5 Recombinações e Mutações

Selecionados os indivíduos, a busca no espaço de soluções é realizada através de dois operadores: reprodução e mutação. A reprodução visa explorar combinações com maior probabilidade para as características das melhores soluções, enquanto a mutação procura aumentar a diversidade das soluções com características novas. Esses operadores são ilustrados na Fig. 3.5.

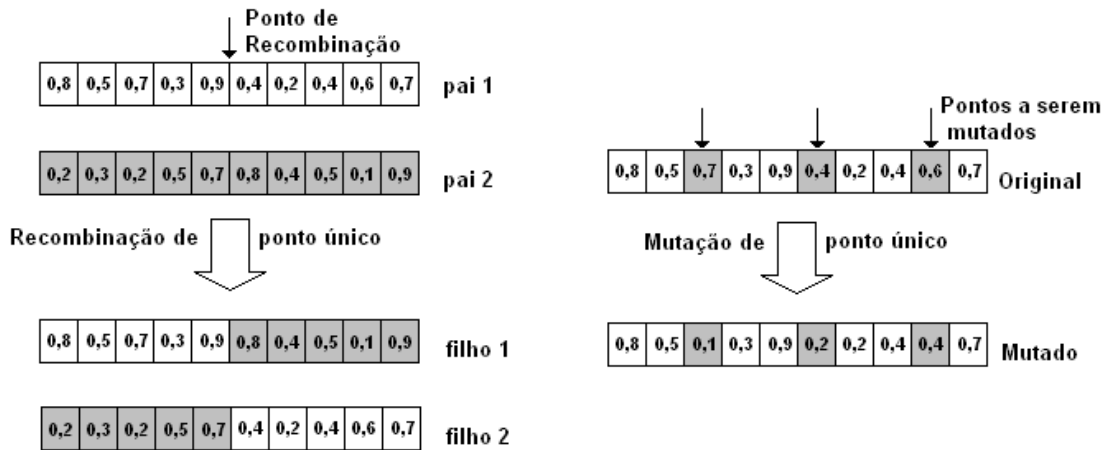


Figura 3.5: Reprodução e Mutação em cromossomos de cadeia real de comprimento $L = 10$.

Na abordagem de Reis (2007), foi proposto um operador de reprodução do tipo aritmético, motivado pela representação por vetores reais (Michalewicz 1996). Esse operador gera um novo indivíduo, I' , como uma combinação linear dos indivíduos *pais* usando um vetor auxiliar β , sendo $\beta_i \in [0, 1]$, buscando explorar no AG os níveis de manutenção de características hereditárias intermediárias em relação aos *pais*.

O operador de mutação gera um novo indivíduo I' através de uma perturbação em cada posição e do vetor real, conforme a Equação 3.4:

$$I_e = \begin{cases} I_e^i + \cup(-\Delta \ \Delta) & \text{se } \tau \leq \rho \\ I_e^i & \text{caso contrário} \end{cases} \quad \forall e \in E, \quad (3.4)$$

sendo τ um número aleatório no intervalo $[0, 1)$, ρ a taxa de mutação (parâmetro do AG que determina a probabilidade da ocorrência de mutação), Δ define a variação máxima permitida, E é o conjunto de elementos sujeitos à manutenção no problema e $\cup(-\Delta, \Delta)$ um valor aleatório no intervalo $(-\Delta, \Delta)$.

Como, ao longo da execução do AG, o intervalo de valores de investimento para cada elemento vai diminuindo, conforme as soluções vão convergindo na população, essa estratégia de mutação busca aumentar os intervalos, garantindo a diversidade na população.

3.3.6 Atualização das Populações

O tipo mais utilizado para a atualização da população em um AG é a *generational* (Michalewicz 1996), onde toda população é substituída por novos indivíduos a cada iteração. Essa estratégia pode descartar boas soluções; por isso Reis (2007) utiliza a estratégia *steady-state*. Nessa estratégia, um ou dois indivíduos são gerados a cada iteração, concorrendo com seus *pais* para serem incluídos na população. Essa estratégia evita a perda de indivíduos bons, porém há o risco da população perder diversidade rapidamente. Para diminuir esse risco, Reis (2007) impede que indivíduos duplicados sejam inseridos na população.

3.3.7 Critério de Parada

O critério de parada determina quando a busca deve ser encerrada. Essa condição pode ser determinada por um número máximo de iterações do processo, ou ainda, por alguma condição que verifique a convergência da população. A convergência pode ser determinada pelo nível de diversidade da população ou apenas medindo o progresso do método ao longo de um número pré-definido de iterações.

Na abordagem de Reis (2007), o critério de parada é o número de iterações ou a convergência baseada no acompanhamento do progresso da busca — o que ocorrer primeiro.

3.3.8 Parâmetros

Os AG's possuem um conjunto de parâmetros para adequar suas características às instâncias específicas do problema em estudo.

Na metodologia proposta por Reis (2007), os parâmetros são determinados empiricamente para cada conjunto de casos estudados. São eles: o tamanho da população, a taxa de mutação, o número máximo de iterações, o progresso mínimo do AG e o tamanho da população inicial na heurística construtiva.

3.4 Hibridização do AG para Alocação de Recursos de Manutenção

Pode-se incorporar nos AG's buscas locais que usam conhecimento no domínio do problema. Essa classe de AG's é usualmente chamada de Algoritmos Genéticos Híbridos (AGH).

Na abordagem de Reis (2007), foi utilizada uma busca para a exploração local do espaço de soluções, caracterizando um AGH representado no Pseudo-código 2.

Para evitar um baixo desempenho do AG, Reis (2007) utiliza uma busca local parcial no espaço de soluções, que limita a exploração local a um determinado número de iterações.

Os principais passos presentes na BL incluída no Pseudo-código 2 procuram reduzir os custos das soluções, realocando o investimento entre pares de elementos em ações mais efetivas, que contribuam para a redução do FEC. Esses passos são apresentados nos Pseudo-códigos 3 e 4. Contudo, a BL não procura realizar investimentos entre todos os pares de elementos, mas apenas estuda os elementos com certa característica de desvio. Deste modo, tem-se uma redução do espaço de busca da BL, diminuindo o número de pares a serem testados de $|E| * |E|$ para $|E'| * |E'|$, onde E' é um subconjunto de E formado pelos elementos $e \in E$ tais que $I_e^1 \neq I_e^2$ com $I_e^1 > 0$ e $I_e^2 > 0$, sendo I_e^1 e I_e^2 o investimento alocado ao elemento e pelas soluções pai 1 e 2, respectivamente.

3.5 Análise do AG para Manutenção

Algumas observações sobre o funcionamento do algoritmo são realizadas a seguir.

- **A complexidade da decodificação**

A decodificação é um processo necessário para um indivíduo que é representado por números reais. A complexidade de sua implementação e o processo de escolha das ações programadas para manutenção podem dificultar a criação de uma população inicial com indivíduos factíveis.

- **A busca pela diversidade da população e convergência prematura**

Por diversas vezes o algoritmo se depara com a dificuldade de produzir diversidade na população. O operador de reprodução aritmético e o processo de escolha dos indivíduos que participam da reprodução são estratégias que buscam combater a falta de diversidade.

- **O risco de atração local**

O risco de uma convergência prematura caminha lado a lado com a perda de diversidade. Sendo assim, o algoritmo evita que indivíduos duplicados sejam inseridos na população.

- **Complexidade da busca local em relação a diversidade da população**

A busca local inserida no AGH explora somente elementos que possuam algum investimento alocado. Essa estratégia produz pouca diversidade, podendo determinar uma convergência prematura.

Essas observações motivaram a investigação de uma nova metodologia, baseada em busca em espaços de estados.

Pseudo-código 2 Algoritmo Genético Híbrido

```

1:  $t \leftarrow 0$ 
2: inicializar  $P(t)$ 
3: avaliar  $P(t)$ 
4: enquanto não atingir a condição de parada faça
5:    $t \leftarrow t + 1$ 
6:   selecionar  $P(t)$  de  $P(t - 1)$ 
7:   realizar cruzamentos entre os indivíduos selecionados de  $P(t)$ 
8:   realizar mutações em indivíduos de  $P(t)$ 
9:   realizar a busca local em  $P(t)$ 
10:  avaliar  $P(t)$ 
11: fim enquanto

```

Pseudo-código 3 Busca Local Parcial (*solucao*)

```

1: inicializar  $E'$ 
2: para  $\forall e \in E'$  faça
3:   para  $\forall f \in E'$  faça
4:      $mov \leftarrow buscarMelhorMovimento(e, f)$  (Pseudo-código 4, a seguir)
5:     se  $mov$  melhora a solução solucao então
6:        $solucao \leftarrow executarMovimento(solucão, mov)$ 
7:        $melhorou \leftarrow true$ 
8:     fim se
9:   fim para
10: fim para
Return: solucao

```

Pseudo-código 4 buscarMelhorMovimento (e, f)

```

1: para  $ano \leftarrow 1$  até  $HP$  faça
2:    $N \leftarrow$  conjunto de níveis de manutenção inferiores ao nível atual do elemento  $e$ 
3:    $M \leftarrow$  conjunto de níveis de manutenção igual ou superiores ao nível atual do elemento  $f$ 
4:   para  $\forall n \in N$  e  $\forall m \in M$  faça
5:      $mov \leftarrow testaMovimento(n, m)$ 
6:      $atualizarMelhorMovimento(mov, melhorMov)$ 
7:   fim para
8: fim para
Return: melhorMov

```

3.5.1 Propostas de Metodologias

A busca em profundidade e a busca em largura são as principais referências nos estudos de busca sem informação. O método *Simulated Annealing* possui como principal característica o mecanismo de escolha de soluções com valores piores, que permite a elaboração de processos de transição aleatórios.

Alguns aspectos a serem explorados na busca em espaços de estados para abordar o problema de otimização de recursos de manutenção são resumidos a seguir.

- **Representação dos indivíduos por números inteiros**

Essa representação é usada por permitir uma forma mais direta de escolha de ações de manutenção. Ainda, o cálculo do FEC pode ser efetuado diretamente, sem precisar de decodificação.

- **Dois tipos de função de Custo/benefício**

O fato do FEC ser uma função cumulativa no horizonte de planejamento motivou um estudo sobre as características implícitas da formulação de duas novas funções Custo/benefício, adequadas ao valor do FEC_{perm} numa determinada solução.

- **Combinação de métodos de buscas**

Esse desenvolvimento foi motivado pela observação das limitações da busca local executada no algoritmo proposto pelo Reis (2007), que procura explorar o espaço de estados de forma mais completa, realizando uma heurística construtiva seguida de uma busca inicial para a confecção de uma boa solução de início do algoritmo. Em seguida, é implementada uma busca específica com informações sobre o problema.

No próximo capítulo são apresentados os conceitos de busca em espaço de estados. Esses conceitos são especializados e detalhados no Capítulo 5, para a abordagem do problema de otimização de estratégias de manutenções para sistemas de distribuição de energia elétrica.

Estratégia de Busca no Espaço de Soluções

4.1 Introdução

O ‘espaço de soluções’ pode ser caracterizado como um ‘espaço de estados’, a expressão mais usual na área de inteligência artificial — busca no espaço de estados é uma das técnicas mais utilizadas para resolução de problemas em Inteligência Artificial (Russell & Norvig 2003).

Neste capítulo, são apresentadas algumas das principais buscas em espaço de estados, com a finalidade de introduzir os conceitos que serão usados no desenvolvimento das meta-heurísticas híbridas.

A descrição das estratégias de busca é apresentada em duas partes. A primeira parte discute buscas sem informação, no sentido de que não usam informação sobre o problema, além de sua definição. A segunda parte discute as buscas com informação, que procuram usar informações sobre o problema na escolha de alternativas para explorar o espaço de busca.

Na primeira parte são apresentados dois tipos de busca no espaço de estados: Busca em profundidade e busca em largura, que têm caráter revogável no processo de transição de estados.

Na segunda parte são apresentados dois tipos de estratégias de busca: o algoritmo *Hill Climbing*, que realiza a busca com caráter irrevogável no processo de transição de estados e o algoritmo de *Simulated Annealing*, também uma estratégia de busca irrevogável, mas com mecanismos de decisão com perturbações estocásticas que procuram melhorar a possibilidade de encontrar boas soluções.

4.2 Busca em Espaço de Estados

A estratégia da busca no espaço de estados supõe a existência de um estado inicial, um conjunto de ações que podem ser executadas para realizar transições de estados e um estado

meta. A solução para esse problema consiste em achar uma sequência de ações que, partindo de um estado inicial, leve a um estado meta. Um espaço de estados é então definido pelo conjunto de estados e pelo conjunto de ações que levam um estado para outro.

Um problema de busca pode ser especificado por quatro componentes:

- a estrutura dos estados;
- o estado inicial;
- o conjunto de estados meta e
- as regras de transição.

A solução de um problema é uma sequência de ações que leva do estado inicial até um estado meta, por uma sequência de ações programadas. As estratégias de busca estudadas nesse capítulo são procedimentos formais que visam encontrar a solução do problema.

O problema de alocação ótima de recursos de manutenção para redes de distribuição de energia elétrica pode ser representado por busca em espaço de estados. Essa representação é exemplificada neste capítulo e desenvolvida apenas no próximo capítulo, com a construção do espaço de estados e dos operadores apropriados para o problema.

Seja $\mathbf{S}_{m \times n}$ uma matriz, onde m é o número de anos definidos para o horizonte de planejamento em um dado problema de alocação ótima de recurso de manutenção e n representa o número de equipamentos de uma rede. A matriz \mathbf{S} representa um estado do problema, ou seja, uma solução para o planejamento da manutenção. Por exemplo, \mathbf{S}_0 pode ser uma matriz que representa um estado inicial para esse problema, a primeira solução. Para um exemplo onde a rede possui 30 equipamentos e o horizonte de planejamento definido é de três anos, \mathbf{S}_0 pode ser construído da seguinte forma:

$$\mathbf{S}_0 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix},$$

onde 1 representa a ação *com manutenção preventiva* e 0 a ação *sem manutenção preventiva*.

Para um problema com apenas 2 decisões possíveis (fazer ou não fazer manutenções), as soluções estão definidas em um espaço de solução de tamanho $2^{(n*m)}$, onde o estado meta é um planejamento de manutenção para o problema que fornece o menor valor de custo, sob restrições de confiabilidade.

Uma transição de estado pode ser representada por uma ação definida da seguinte forma:

$$\text{tipodeaçãoescolhido}(\mathbf{S}, \mathbf{S}')$$

onde \mathbf{S} é o estado atual e \mathbf{S}' é o estado sucessor fornecido pela ação escolhida ($\mathbf{S} \rightarrow \mathbf{S}'$).

4.2.1 Busca em Profundidade

Na busca em profundidade, a prioridade é buscar o nó de nível mais profundo da árvore de busca (Pearl 1984). A estratégia da busca em profundidade é expandir o nó analisado, explorando primeiro os nós gerados (filhos) por esse nó antes que os outros nós encontrados (irmãos) sejam analisados. Ou seja, todos os nós descendentes do nó expandido são explorados antes de seus irmãos. Considere o grafo representado na Fig.4.1 como um exemplo de problema para a aplicação da busca em profundidade (Luger & Stubblefield 1993).

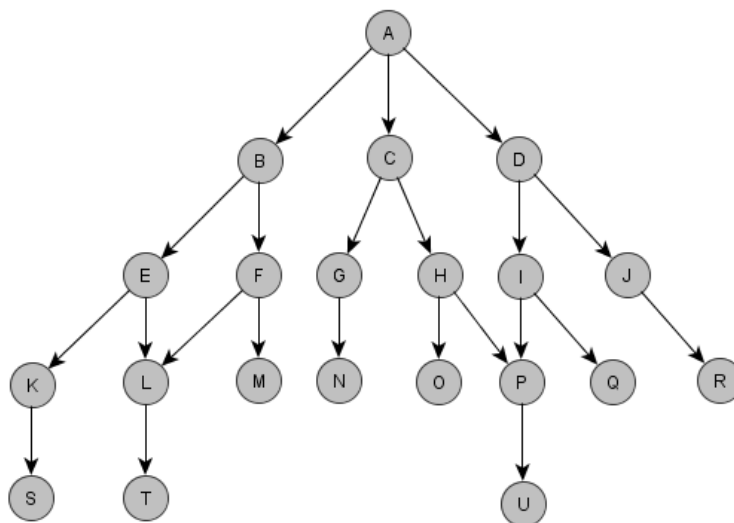


Figura 4.1: Exemplo de Grafo para a aplicação da Busca em Profundidade.

Seguindo a estratégia descrita, a ordem em que os nós são visitados é A, B, E, K, S, L, T, F, M, C, G, N, H, O, P, U, D, I, Q, J, R. Essa estratégia pode ser implementada com uma estrutura computacional *last-in-first-out* (LIFO), também conhecida como PILHA (Russell & Norvig 2003).

Na busca-em-árvore o conceito de profundidade é bem definido. Portanto, um algoritmo de busca em profundidade não encontra dificuldades na escolha dos próximos nós a serem explorados. Como o algoritmo usa uma estrutura de PILHA, a estratégia da profundidade é colocar os novos sucessores no topo dessa PILHA e escolher para a expansão o nó que estiver no topo. A implementação dessa estratégia é representada no Pseudo-código 5 (Luger & Stubblefield 1993).

A busca é implementada utilizando os conjuntos Abre (*Open*) e Fecha (*Closed*), estruturas na forma de pilha. Os nós explorados são selecionados e retirados pela extrema esquerda no conjunto Abre e colocados à esquerda no conjunto Fecha, e seus filhos são inseridos à esquerda do conjunto Abre. Esse procedimento segue para todos os descendentes, até que a

busca encontre um nó objetivo. Se não for encontrado um nó objetivo, o algoritmo retorna uma indicação de falha.

Pseudo-código 5 Algoritmo - Busca em Profundidade

```

1: Abre:= [início];
2: Fecha:= [];
3: Enquanto Abre ≠ vazio faça
4:   remova o extremo da esquerda de Abre e chame de X
5:   Se X for um objetivo faça
6:     Retorna: Sucesso
7:   Senão
8:     gere os filhos de X
9:     insira X em Fecha
10:    elimine os filhos que já estiverem em Abre ou Fecha
11:    insira os filhos restantes à esquerda em Abre
12:   Fim Se
13: Fim Enquanto
Retorna: Falha

```

A aplicação do procedimento executado pelo algoritmo de busca em profundidade no exemplo de grafo representado na Fig.4.1 é resumida na sequência definida a seguir.

1. $Abre = [A]; Fecha = []$
2. $Abre = [B,C,D]; Fecha = [A]$
3. $Abre = [E,F,C,D]; Fecha = [B,A]$
4. $Abre = [K,L,F,C,D]; Fecha = [E,B,A]$
5. $Abre = [S,L,F,C,D]; Fecha = [K,E,B,A]$
6. $Abre = [L,F,C,D]; Fecha = [S,K,E,B,A]$
7. $Abre = [T,F,C,D]; Fecha = [L,S,K,E,B,A]$
8. $Abre = [F,C,D]; Fecha = [T,L,S,K,E,B,A]$
9. $Abre = [M,C,D]$, já que L estava em Fecha; $Fecha = [F,T,L,S,K,E,B,A]$
10. $Abre = [C,D]; Fecha = [M,F,T,L,S,K,E,B,A]$
11. $Abre = [G,H,D]; Fecha = [C,M,F,T,L,S,K,E,B,A]$
12. Até que o nó objetivo seja encontrado, ou $Abre=vazio$

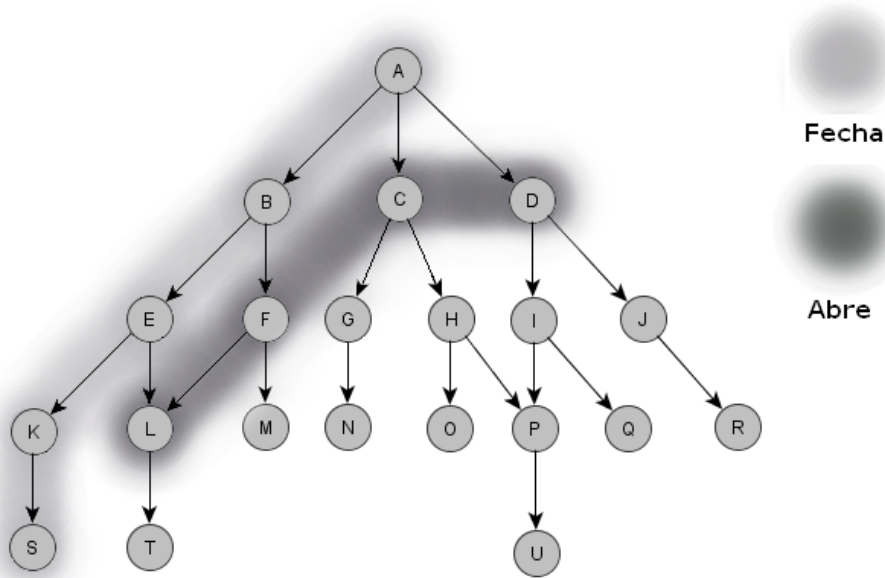


Figura 4.2: Grafo da Fig 4.1 na iteração seis do algoritmo de busca em profundidade.

A Fig.4.2 ilustra a Etapa 6, destacando os conjuntos Abre e Fecha (Luger & Stubblefield 1993).

A busca em profundidade utiliza poucos recursos de memória, precisando guardar apenas o caminho dado pela raiz até o nó folha (nó que não possui sucessores) e os irmãos não-explorados de cada nó no caminho. Um nó pode ser removido da memória depois que todos seus sucessores forem explorados. Sendo assim, para um espaço de estados com fator de ramificação b e profundidade máxima m , a busca armazenará apenas $(bm + 1)$ nós.

O risco da busca em profundidade entrar em um processo ininterrupto de geração de nós sucessores pode ser diminuído com a atribuição de um limite de profundidade determinado, l . Os nós de profundidade l são tratados como nós folha, reduzindo o armazenamento para $(bl + 1)$ e definindo a busca como *Busca em Profundidade Limitada*. Porém, se o tamanho de l for menor que a profundidade do nó objetivo mais raso d a busca será incompleta.

A busca em profundidade explorada nas metodologias desenvolvidas nesta dissertação possui uma função de avaliação para a escolha dos filhos, ou seja, ela inclui características de uma busca informada.

4.2.2 Busca em Largura

Na Busca em Largura todos os nós do nível atual são expandidos antes de qualquer nó do nível seguinte. A estratégia segue expandindo todos os nós desse nível seguinte, e depois os nós dos níveis sucessores a esse nível, e assim por diante.

A estratégia pode ser implementada por uma estrutura computacional *first-in-first-out*

(FIFO), conhecida como FILA (Russell & Norvig 2003). Os sucessores recém-chegados são colocados no final da FILA, assegurando que os nós de baixa profundidade serão expandidos antes dos nós mais profundos.

Considere o grafo da Fig.4.3, o mesmo utilizado na seção anterior (Luger & Stubblefield 1993). Dada a estratégia descrita, a ordem em que os nós do grafo são visitados é A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U. A estratégia pode ser ilustrado através do algoritmo de busca em largura representado no Pseudo-código 6 (Luger & Stubblefield 1993), implementado usando os conjuntos Abre (Open) e Fecha (Closed). O conjunto Abre possui uma FILA.

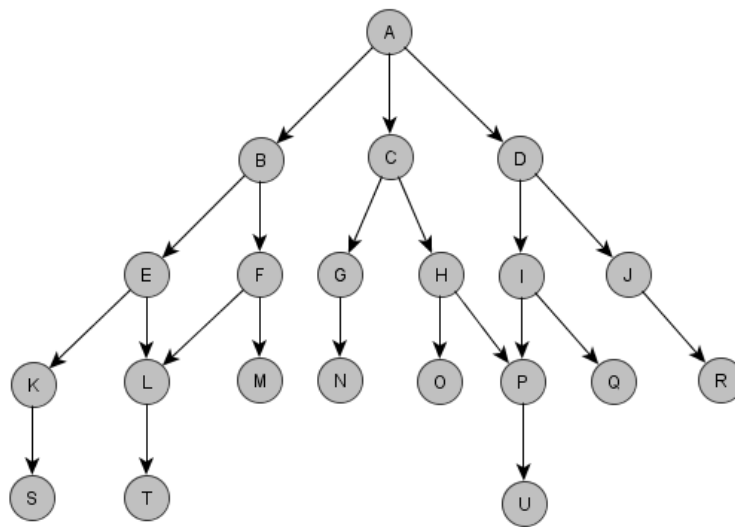


Figura 4.3: Exemplo de Grafo para aplicação da Busca em Largura.

A ordem em que os nós são removidos do conjunto Abre determina a sequência de busca. O nó da extrema esquerda é explorado, removido de Abre e inserido à esquerda em Fecha. Seus filhos gerados, depois de serem eliminados os repetidos, são inseridos à direita em Abre. A busca termina quando um nó objetivo é encontrado, retornando uma indicação de sucesso, ou quando Abre estiver vazio, retornando um indicação de falha.

Dado o grafo da Fig.4.3, o conteúdo das estruturas Abre e Fecha durante a aplicação do algoritmo são mostrados a seguir.

1. Abre = [A]; Fecha = []
2. Abre = [B,C,D]; Fecha =[A]
3. Abre = [C,D,E,F]; Fecha =[B,A]

Pseudo-código 6 Algoritmo - Busca em Largura

```

1: Abre:= [início];
2: Fecha:= [];
3: Enquanto Abre ≠ vazio faça
4:   remova o extremo da esquerda de Abre e chame de X
5:   Se X for um objetivo faça
6:     Retorna: sucesso
7:   Senão
8:     gere os filhos de X
9:     insira X em Fecha
10:    elimine os filhos que já estiverem em Abre ou Fecha
11:    insira os filhos restantes à direita em Abre
12:   Fim Se
13: Fim Enquanto
Retorna: Falha

```

4. $Abre = [D,E,F,G,H]; Fecha = [C,B,A]$
5. $Abre = [E,F,G,H,I,J]; Fecha = [D,C,B,A]$
6. $Abre = [F,G,H,I,J,K,L]; Fecha = [E,D,C,B,A]$
7. $Abre = [G,H,I,J,K,L,M]$ já que *L* estava em *Fecha*; $Fecha = [F,E,D,C,B,A]$
8. $Abre = [H,I,J,K,L,M,N]; Fecha = [G,F,E,D,C,B,A]$
9. Até que o nó objetivo seja encontrado, ou $Abre = \text{vazio}$

A Fig.4.4 ilustra a Etapa 6 da sequência de busca, destacando os nós pertencentes aos conjuntos *Abre* e *Fecha* (Luger & Stubblefield 1993).

Desde que o número máximo de sucessores de um certo nó (fator de ramificação b) seja finito, pode-se dizer que a busca em largura sempre alcança um nó meta, desde que o nó exista. Se o nó objetivo mais raso estiver em alguma profundidade finita d , a busca o encontrará após expandir todos os nós com profundidade menor que d .

O tempo de execução e a memória empregada na busca em largura podem ser grandes. Considerando que cada nó expandido gere b sucessores, a cada nível explorado são acrescentados b sucessores a cada nó (Russell & Norvig 2003). Se d é o último nível então o número total de nós gerados é $b + b^2 + b^3 + \dots + b^d + (b^{d+1} - b)$. Todo nó gerado permanece na memória, pois é um nó que pertence ao último nível ou é um ancestral desse nó.

A busca em largura desenvolvida para o problema de manutenção de redes usa uma função de avaliação para escolher os nós sucessores, além de uma limitação imposta no números de nós gerados. Essas estratégias procuram diminuir o tempo de processamento e os recursos de memória utilizados pela busca, ou seja, a busca em largura desenvolvida adota conceitos de uma busca informada.

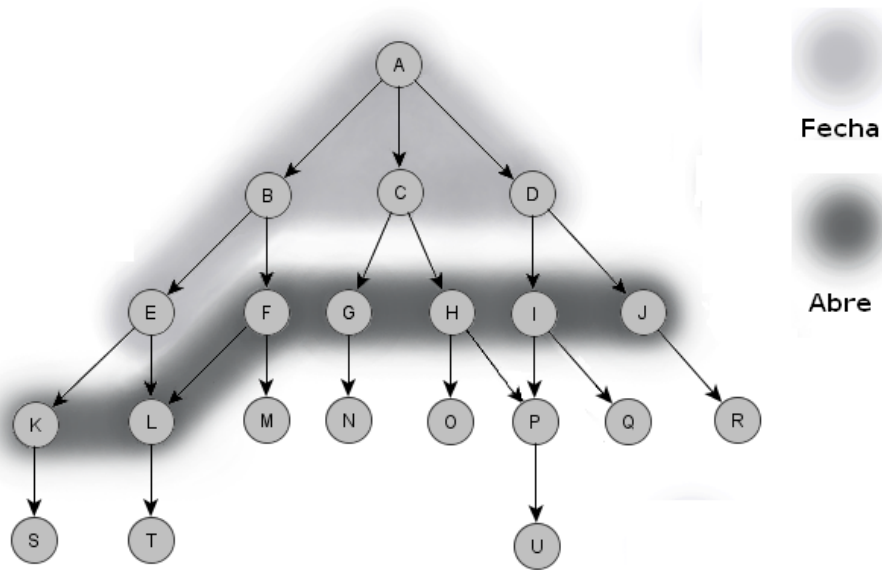


Figura 4.4: Grafo da Fig 4.3 na iteração seis do algoritmo de busca em largura.

4.2.3 Hill Climbing

O método *Hill Climbing*¹ usa uma função de avaliação para escolher o próximo estado a ser explorado. O método é chamado também de busca gulosa, pois a cada passo procura a maior melhoria na função avaliação. A estratégia termina quando um valor da função, associado a um certo estado não puder ser melhorado em estados vizinhos. É uma busca irrevogável (Pearl 1984), pois não possui mecanismos para retroceder a estados anteriores.

Os riscos que a busca gulosa possui são os de produzir pontos com as seguintes características:

- Ótimos locais, ou seja, pontos em uma região de atração que possuem valores melhores que todos os seus estados vizinhos, mas sendo piores que o ótimo global;
- Picos, formados por uma sequência de ótimos locais, o que determina um caminho mais tortuoso para a busca;
- Platôs, ou seja, uma área do espaço de estados onde a função de avaliação é plana.

O Pseudo-código 7 resume os passos da estratégia *Hill Climbing*. A função valor, na quarta linha do algoritmo, guia o processo de busca no espaço de soluções.

O método *Hill Climbing* na forma padrão é uma busca irrevogável que nunca executa movimentos que piorem os valores dados pela função de avaliação. Por isso, a busca explora apenas certas regiões de espaço de buscas, o que a torna uma busca incompleta. Seus requisitos de memória são pequenos no caso padrão, pois armazena apenas o ponto atual. Seu

¹Pode ser traduzido por Escalada de Encosta

Pseudo-código 7 Algoritmo Hill Climbing Padrão

```

1: Inicializa  $s_{inicial}$  aleatoriamente
2:  $s \leftarrow s_{inicial}$ 
3: Repita
4:   Encontra  $s'$  vizinho
5:   Se  $valor(s')$  for melhor que  $valor(s)$  então
6:      $s \leftarrow s'$ 
7:   Fim Se
8: Até que não se encontre vizinhos melhores ou número de iterações
Retorna:  $s$ 

```

tempo de processamento também é pequeno, consequência do fato da busca explorar uma região limitada de espaço de estados.

Além do método *Hill Climbing* padrão foram criadas variantes:

- *Hill Climbing* Iterativo;
- *Hill Climbing* Estocástico.

O método de *Hill Climbing* Iterativo faz a busca a partir de pontos iniciais gerados aleatoriamente, procurando melhorar a solução encontrada até o momento. O Pseudo-código 8 resume o algoritmo.

Pseudo-código 8 Algoritmo Hill Climbing Iterativo

```

1: Inicializa  $s_{melhor}$  como a melhor solução até o momento
2: Para cada condição inicial definida faça
3:   Inicializa  $s_{inicial}$  aleatoriamente
4:   Repita
5:     Encontra  $s'$  vizinho
6:     Se  $valor(s')$  for melhor que  $valor(s)$  então
7:        $s \leftarrow s'$ 
8:     Fim Se
9:   Até que não se encontre vizinhos melhores ou número de iterações
10: Fim Para
11: Se  $valor(s)$  for melhor que  $valor(s_{melhor})$  então
12:    $s_{melhor} \leftarrow s$ 
13: Fim Se
Retorna:  $s_{melhor}$ 

```

O método de *Hill Climbing* Estocástico fornece uma outra alternativa para exploração de condições iniciais diferentes. Sua principal característica é a escolha de um vizinho de forma aleatória, com uma probabilidade p_h . A probabilidade de escolha p_h depende da qualidade relativa, definida pelos valores da função de avaliação aplicada aos pontos:

$$p_h = 1/(1 + (\exp((val(s) - val(s'))/T))), \quad (4.1)$$

onde s é o ponto inicial, s' é o vizinho escolhido, $val(s)$ é o valor da função avaliada no ponto s , $val(s')$ é o valor da função avaliada no ponto s' e T é uma constante. O Pseudo-código 9 resume os passos de busca *Hill Climbing* Estocástico.

Pseudo-código 9 Algoritmo *Hill Climbing* Estocástico

```

1: Inicializa  $s_{inicial}$  aleatoriamente
2: Repita
3:   Encontra  $s'$  vizinho
4:   Se  $p_h < valor\ definido$  então
5:      $s \leftarrow s'$ 
6:   Fim Se
7: Até que não se encontre vizinhos melhores ou número de iterações
Retorna:  $s$ 

```

Para fazer com que a busca *Hill Climbing* sempre encontre um estado meta, os movimentos realizados teriam que ser puramente aleatórios, o que tornaria a busca muito ineficiente. Uma combinação do método de *Hill Climbing* com movimentos aleatórios pode melhorar o processo na direção de uma busca completa com eficiência. Essa combinação caracteriza a busca *Simulated Annealing*.

Um outro tipo de método que realiza uma busca gulosa é a heurística GRASP (greedy randomized adaptive search procedure) (Feo & Resende 1995). O GRASP é um método que combina uma fase construtiva com uma busca local. Na fase construtiva, elabora-se uma busca gulosa com aleatoriedade para a confecção da primeira solução. Na busca local, realiza uma investigação na vizinhança, até que se encontre uma melhora para essa solução encontrada na fase construtiva. Uma metodologia que utiliza o método GRASP foi implementada por Reis (2007) para o problema de alocação ótima de recursos de manutenção em redes de distribuição de energia elétrica. Porém, essa metodologia não será objeto de estudo nessa dissertação, pois os experimentos realizados nos estudos de casos desenvolvido por Reis (2007), apresentaram piores resultados quando comparados ao algoritmo genético híbrido, também desenvolvido por Reis (2007) e apresentado no capítulo anterior.

4.3 Simulated Annealing

O *Simulated Annealing*² foi baseado no procedimento de Metropolis, Rosenbluth, Rosenbluth, Teller & Teller (1953), proposto como uma estratégia de determinação de estados de equilíbrio de uma coleção de átomos a uma dada temperatura. O procedimento dos autores continha dois passos (Aarts & Korst 1989):

- Elevar a temperatura do cozimento ao máximo para o processo de ‘derretimento’ do sólido ocorra;

²Pode ser traduzido como Recozimento Simulado

- Diminuir a temperatura do cozimento cuidadosamente até que o sólido volte ao seu estado inicial.

A estratégia de *Simulated Annealing* proposta por Kirkpatrick, Gerlatt & Vecchi (1983) foi inspirada pelo processo de recozimento de sistemas físicos. Eles perceberam que existe uma semelhança entre o procedimento de recozimento estudado com o algoritmo de Metropolis et al. (1953) e processos de otimização combinatorial. A Tabela 4.1 ilustra a relação entre a estratégia proposta por Kirkpatrick et al. (1983) e o processo de recozimento de sistemas físicos de Metropolis et al. (1953).

Termodinâmica	Algoritmo de Simulated Annealing
Estado (configuração)	Solução (factível) do problema
Energia	Valor retornado pela função de avaliação
Estado de equilíbrio	Ótimo (local)
Estado de equilíbrio máximo	Ótimo global
Temperatura	Parâmetro de controle
Recozimento	Busca através da redução de T
Distribuição de Boltzmann-Gibbs	Probabilidade de seleção de um novo ponto

Tabela 4.1: Correspondência entre a Interpretação da Terminologia do Sistema Físico e o Domínio Computacional.

A estrutura do algoritmo do *Simulated Annealing* é semelhante a do *Hill Climbing* Estocástico, onde alguns movimentos de piora do valor da função avaliada são permitidos. Assim, o laço de repetição do algoritmo é semelhante ao do *Hill Climbing*, diferenciando no modo de escolha do movimento; enquanto o *Hill Climbing* busca sempre o melhor movimento, *Simulated Annealing* permite movimentos aleatórios.

A execução do movimento é baseada em uma probabilidade que vai diminuindo exponencialmente, à medida em que um movimento ruim é executado. Se o movimento realizado melhorar a função de avaliação, ele sempre será aceito. Caso contrário, o algoritmo aceitará o movimento com uma probabilidade $p < 1$.

A probabilidade p depende da qualidade relativa entre os valores da função de avaliação quando aplicada aos pontos e da variável de ‘temperatura’ T :

$$p = 1/(1 + (\exp((val(s) - val(s'))/T))), \quad (4.2)$$

onde s é o ponto inicial, s' é ponto do movimento seguinte, $val(s)$ é o valor da função avaliada no ponto s , $val(s')$ é o valor da função avaliada no ponto s' e T é a variável ‘temperatura’. O Pseudo-código¹⁰ resume o algoritmo de *Simulated Annealing*.

A variável T tem uma característica importante na funcionalidade do algoritmo. Quanto mais devagar o valor de T for diminuindo maior será a chance do algoritmo encontrar um ótimo global (Russell & Norvig 2003), mas o tempo computacional da busca aumentará.

Pseudo-código 10 Algoritmo Simulated Annealing

```

1: Inicializa  $s_{inicial}$  aleatoriamente
2: Inicializa  $T$ 
3: Repita
4:   Encontra  $s'$  movimento seguinte
5:   Se  $valor(s') < valor(s)$  então
6:      $s \leftarrow s'$ 
7:   Senão
8:     Se  $p < valor\ definido$  então
9:        $s \leftarrow s'$ 
10:     $T$  diminui (esfria)
11:  Fim Se
12: Fim Se
13: Até que não se encontre vizinhos melhores ou número de iterações
Retorna:  $s$ 

```

Pode-se dizer que algoritmo de *Simulated Annealing* tem uma probabilidade maior de encontrar um estado meta do que o algoritmo de *Hill Climbing*.

As exigências de memória da busca *Simulated Annealing* são pequenas, pois o algoritmo só guarda o movimento atual. Porém, seu tempo computacional é grande, comparado a outros métodos de busca (Goffe, Ferrier & Rogers 1994), sendo bem maior que o tempo computacional do *Hill Climbing* padrão, pois executa movimentos que pioram o valor da função de avaliação.

4.4 Comentários

Outros tipos de buscas com informação poderiam ser explorados na busca no espaço de estados, entres eles figuram:

- Algoritmo de *Backtracking*:

A estratégia é listar os possíveis caminhos a serem escolhidos procurando eliminar através de um mecanismo de avaliação, as alternativas pouco promissoras. Pode ser considerada como uma variação da busca em profundidade.

- Algoritmo A*:

É uma estratégia de busca informada (pois procura informações sobre o problema) que determina a expansão dos nós através da indicação de uma função de avaliação. Essa função combina $g(n)$, o custo para alcançar um certo nó n , e $h(n)$, uma estimativa do custo pra ir do nó n até o nó objetivo:

$$f(n) = g(n) + h(n) \quad (4.3)$$

Essas buscas têm algumas características semelhantes às buscas apresentadas nesse capítulo. A avaliação de eventuais benefícios de suas utilizações nos problemas estudados podem vir a ser realizadas em investigações futuras.

As buscas em profundidade, largura e *Hill Climbing*, são as principais referências para buscas em espaços de estados. Por esse motivo, foram referências para as investigações realizadas para o desenvolvimento de meta-heurísticas híbridas, discutidas no próximo capítulo.

As extensões elaboradas da busca em profundidade e da busca em largura, citadas nesse capítulo, combinadas com o método de *Simulated Annealing* produzem diferentes tipos de meta-heurísticas, desenvolvidas e detalhadas no próximo capítulo. Nessas buscas em espaço de estados, diversos tipos de ações de transição de estados são explorados.

Meta-heurísticas Híbridas

5.1 Introdução

As metodologias propostas neste trabalho possuem diferentes estruturas e são dotadas de estratégias, combinando técnicas de Busca em Profundidade, Busca em Largura e *Simulated Annealing*. As metodologias são apresentadas usando pseudo-códigos, seguidas pelos respectivos fluxogramas. Os processos de construção da primeira solução e das buscas no espaço de soluções também são discutidos.

5.2 Buscas no Espaço de Estados para o Problema de Manutenção

As abordagens foram desenvolvidas conforme as ideias de melhorias propostas no Capítulo 3. Tratam-se de novos procedimentos que buscam reduzir o tempo de processamento e melhorar o custo da solução final encontrada para o problema de manutenção de SDEE. Estratégias de busca no espaço de estados são combinadas para formar procedimentos listados a seguir.

Procedimento 1 - Busca em Profundidade com *Simulated Annealing* (BPS)

- Heurística Construtiva (HC)
- Busca Inicial - Busca em Profundidade (BI)
- *Simulated Annealing* (SA)

Procedimento 2 - Busca em Largura com *Simulated Annealing* (BLS)

- Heurística Construtiva

- Busca Inicial - Busca em Largura
- *Simulated Annealing*

Procedimento 3 - Busca *Simulated Annealing* (BSA)

- Heurística Construtiva
- *Simulated Annealing*

A Heurística Construtiva é a mesma para todas os procedimentos. Sua função é reduzir o número de iterações do algoritmo, elaborando uma solução inicial de boa qualidade para o processo. A Busca Inicial definida pelos Procedimentos 1 e 2, tem a função de encontrar uma solução melhorada a partir da solução inicial obtida pela HC. O *Simulated Annealing* é o principal processo de busca em todos os procedimentos, realizando a busca por melhores soluções usando informações do problema.

5.3 A Estrutura das Novas Abordagens

Os procedimentos citados são detalhados à seguir.

5.3.1 Heurística Construtiva

Para a construção da solução inicial, a HC é auxiliada pelo uso de funções custo/benefício (CB) que determinam uma medida de investimento para cada equipamento. Dois tipos de funções CB são elaboradas:

$$CB1(t, e) = \frac{(M_{SM_e} - M_{CM_e})N_e}{C_{MP_e}\alpha_t \exp(t)} e \quad (5.1)$$

$$CB2(t, e) = \frac{(FEC_{perm} - FEC_{min})(M_{SM_e} - M_{CM_e})N_e}{C_{MP_e}\alpha_t}, \quad (5.2)$$

sendo M_{SM_e} o multiplicador da taxa de falha para o equipamento e definido sem manutenção preventiva, M_{CM_e} o multiplicador da taxa de falha para o equipamento e definido com manutenção preventiva, N_e o número de clientes presentes na seção onde o equipamento e pertence, C_{MP_e} o custo de manutenção preventiva prevista para o equipamento e , α_t o parâmetro definido na Subseção 2.4.3 para o período t , FEC_{perm} é o máximo valor de FEC permitido e FEC_{min} o valor de FEC calculado para a formulação quando a qualidade de serviço é máxima.

A elaboração de dois tipos de funções é justificada pelas características presentes no modelo de problema adotado. Quando a restrição de confiabilidade do problema é mais

apertada, a solução final tende a priorizar a manutenção preventiva no primeiro ano de horizonte de planejamento, pois o cálculo da restrição sugere um valor cumulativo ao longo dos períodos. Assim, é necessário distinguir o custo/benefício de cada equipamento ao longo dos anos apenas de forma suave para obter uma boa solução inicial. Quando a restrição de confiabilidade é mais folgada, a solução final tende a priorizar a manutenção preventiva no último ano de horizonte de planejamento, devido ao cálculo da taxa de juros anuais α_t . Deste modo, é necessário distinguir o custo/benefício de cada equipamento ao longo dos anos de forma mais rigorosa. Esse fato justifica o uso da $\exp(t)$, que acentua a diferença entre os valores de custo/benefício dos equipamentos a cada período, fornecendo uma melhor solução inicial para esse problema.

Essas funções são escolhidas de acordo com o valor de FEC_{perm} definido. Se a diferença entre o valor de FEC_{perm} e o valor de FEC_{min} for menor que um valor especificado, escolhe-se CB1, senão a CB2 é escolhida.

Escolhida a função custo/benefício, ela é calculada para todos os elementos de \mathbf{S} , armazenando os valores calculados na matriz MI de medida de investimentos. Em seguida, é obtido o maior valor de medida de investimento, MI_{max} . Esse valor é comparado com cada uma das medidas de investimento calculada. Se a medida de um elemento possuir um valor maior ou igual ao valor da maior medida selecionada, o elemento recebe manutenção. Caso contrário, não recebe manutenção, construindo uma solução inicial.

Testa-se a factibilidade dessa solução inicial. Se ela for factível a HC termina e o algoritmo retorna a solução construída, senão essa estratégia é reiniciada, selecionando a segunda maior medida de investimento em CB. Novamente, as medidas de todos os elementos são comparadas com esse valor e uma nova solução inicial é construída. Esse procedimento segue até que a solução inicial elaborada seja factível, dada a restrição de qualidade de serviço presente na formulação (FEC). O Pseudo-código 11 resume este procedimento.

A estratégia da HC encontra a primeira solução factível, portanto terá um valor de restrição próximo do valor do FEC_{perm} . Deste modo, como foi visto na Fig.2.8 da Seção 2.4.3, quanto menor a confiabilidade, menor o custo da solução, e vice-versa. Isso implica uma redução dos passos de buscas do algoritmo. Porém, essa vantagem pode acarretar uma convergência prematura. Para diminuir esse risco, as buscas iniciais utilizam os métodos de *busca em profundidade* e o método de *busca em largura*, a fim de realizar uma busca sem informação específica, mas explorando as melhores alternativas de soluções.

5.3.2 Busca Inicial

Os procedimentos de BI desenvolvidos utilizam os métodos de busca em profundidade e de busca em largura. Em ambos os métodos, o processo de transição de estados é dado por movimentos aplicados a cada combinação possível de pares de elementos. Esses movimentos,

Pseudo-código 11 Heurística Construtiva (S)

```

1: Escolhe-se CB
2: Para  $\forall t \in HP$  faça
3:   Para  $\forall s \in S$  faça
4:     calcule CB(t,s)
5:      $MI_{t,s} \leftarrow \mathbf{CB}(t,s)$ 
6:   Fim Para
7: Fim Para
8: Repita
9:    $MI_{\max} \leftarrow \max(MI)$ 
10: Para  $\forall t \in HP$  faça
11:   Para  $\forall s \in S$  faça
12:     Se  $MI_{t,s} = MI_{\max}$  então
13:        $MI_{t,s} \leftarrow 0$ 
14:        $S_{t,s} \leftarrow 1$  // tem manutenção
15:     Senão
16:        $S_{t,s} \leftarrow 0$  // sem manutenção
17:     Fim Se
18:   Fim Para
19: Fim Para
20: Até que (S factível)
Retorna: S

```

detalhados a seguir, alteram apenas um par de elementos da solução por iteração do movimento. Os pares são selecionados a partir de uma solução modelo definida através da função CB construída em HC.

Para cada par de elementos, uma nova solução é elaborada e sua factibilidade é testada. Se essa solução for factível, é calculado o valor de custo da função objetivo, ou seja, se o valor de custo dessa solução tiver um resultado melhor que o valor de custo da solução atual, a solução atual é substituída. Caso contrário, o procedimento continua.

Depois que todos os pares selecionados forem analisados e nenhuma solução produzida pelo movimento obtiver um valor melhor que o valor do custo da solução atual, a estratégia do método de busca (profundidade ou largura) é executada.

O término das BIs é dado por dois critérios de parada. O primeiro usa um parâmetro que mede o grau de eficiência do método de busca; o outro usa o número de iterações do processo. O primeiro critério de parada analisa se a pilha para a busca em profundidade, ou a fila para a busca em largura, não está vazia. Caso esteja, o processo é encerrado. O segundo critério de parada, baseado no número de iteração, consiste em terminar a busca quando um número máximo de iterações previamente determinado for atingido. A saída S_{final} é a solução final parcial encontrada pela BI, traduzida como solução inicial para o processo seguinte. Esses processos iterativos são resumidos nos Pseudo-códigos 12 e 13.

Os movimentos efetuam uma busca no espaço de estados na vizinhança da solução inicial.

Pseudo-código 12 Busca Inicial - Profundidade (**S**)

```

1:  $S_{aux} \leftarrow S$ 
2:  $PILHA \leftarrow S$ 
3: Repita
4:    $S_{s'} \leftarrow Movimento\ 1$ 
5:   Seleciona  $S' \in S_{s'}$  com menor valor de custo
6:   Se  $Custo(S') < Custo(S_{aux})$  então
7:      $S \leftarrow S'$ 
8:      $S_{aux} \leftarrow S'$ 
9:     insira  $S'$  na  $PILHA$ 
10:  Se  $Custo(S') < CustoFinal$  então
11:     $S_{final} \leftarrow S'$ 
12:     $CustoFinal \leftarrow Custo(S')$ 
13:  Fim Se
14: Senão
15:    $S_{aux} \leftarrow$  primeiro da  $PILHA$ 
16:   retira o primeiro da  $PILHA$ 
17:  Se  $PILHA \neq$  vazio então
18:     $S \leftarrow$  primeiro da  $PILHA$ 
19:  Senão
20:    Encerra a Busca
21:  Fim Se
22: Fim Se
23: Até que (Limite de Iteração)
Retorna:  $S_{final}$ 

```

Dotados das estratégias de busca em profundidade e de busca em largura, visam melhorar a solução inicial explorando soluções conforme regras de transição descritos a seguir.

Movimento 1

As transição efetuadas nas BIs são baseadas em um único tipo de movimento: operam pares específicos de elementos da solução, alterando a ação de manutenção programada. Esse conceito de alteração do nível de manutenção tem apenas duas versões. Assim, duas ações (bl_{1a} e bl_{1b}) são elaboradas para a operação do movimento. A alteração é válida quando ao menos um deles tem condições de sofrer essas mudanças.

A escolha dos pares para o movimento é determinada por uma solução modelo (S_m) obtida pelo cálculo da média das medidas de investimento, calculada na HC. A média é usada como um valor de referência, definindo as ações de manutenção da solução S_m . Se a medida de investimento de um elemento é maior ou igual à média obtida, então o respectivo elemento da solução modelo recebe manutenção. Caso contrário, não recebe manutenção. A solução S_m é criada, aplicando essa estratégia para todos os elementos em todos os anos.

A realização do movimento em um certo par de elementos é determinada pela solução S_m .

Pseudo-código 13 Busca Inicial - Largura (**S**)

```

1:  $S_{aux} \leftarrow S$ 
2:  $FILA \leftarrow S$ 
3:  $contador \leftarrow 0$ 
4: Repita
5:    $S_{s'} \leftarrow Movimento\ 1$ 
6:   Selecciona  $S' \in S_{s'}$  com menor valor de custo
7:   Se  $Custo(S') \leq Custo(S_{aux})$  e  $contador < 3$  então
8:     Para  $\forall S_{aux} \in S_{s'}$  faça
9:       Se  $Custo(S') = Custo(S_{aux})$  então
10:        insira  $S'$  na  $FILA$ 
11:      Fim Se
12:    Fim Para
13:    retire o primeiro da  $FILA$ 
14:     $S_{aux} \leftarrow primeiro\ da\ FILA$ 
15:     $S \leftarrow primeiro\ da\ FILA$ 
16:    Se  $S_{aux} = S'$  então
17:      incrementa  $contador$ 
18:    Senão
19:       $S_{aux} \leftarrow S'$ 
20:       $contador \leftarrow 0$ 
21:    Fim Se
22:  Senão
23:    retira o primeiro da  $FILA$ 
24:    Se  $FILA \neq vazio$ 
25:       $S \leftarrow primeiro\ da\ FILA$ 
26:    Senão
27:      Encerra a Busca
28:  Fim Se
29: Até que (Limite de Iteração)

```

Retorna: S

Se ambos os elementos do par selecionado na solução atual tiverem ações de manutenção iguais ao elementos do par referente descrito na solução modelo, eles sofrerão alteração no seu nível de manutenção. Caso contrário, esse par de elementos selecionado é ignorado. Para cada aplicação do movimento, uma nova solução é produzida e armazenada em $S_{s'}$. O processo de transição do movimento 1 é ilustrado na Fig.5.1. No Movimento 1, S é a solução atual, S_m é a solução modelo, pos é o operador que define a posição do elemento, os índices i e k representam a linha e os índices j e l representam a coluna da matriz descrita na Seção 4.2; e \oplus é o operador lógico *ou exclusivo*. O processo iterativo da transição de estados com a avaliação da solução sucessora produzida é ilustrado no Pseudo-código 14. A Fig.5.2 ilustra o processo de transição em um determinado par de elementos da solução atual.

$bl_1(\text{ação}, \mathbf{S}, \mathbf{S}_m)$ $\forall p \in P$ <p>Se ação == bl_{1_a} então</p> $\forall s'_a \text{ e } s'_b \in \mathbf{S}'$ $\mathbf{S}' \leftarrow \mathbf{S}$ $s'(a) = 1 \text{ e } s'(b) = 0$ $\mathbf{S}_{s'} \cup \mathbf{S}'$ <p>Senão Se ação == bl_{1_b} então</p> $\forall s'_a \text{ e } s'_b \in \mathbf{S}'$ $\mathbf{S}' \leftarrow \mathbf{S}$ $s'(a) = 0 \text{ e } s'(b) = 1$ $\mathbf{S}_{s'} \cup \mathbf{S}'$ $P \leftarrow \text{seldepare}(\mathbf{S}, \mathbf{S}_m)$ $\forall s \in \{\mathbf{S} \oplus \mathbf{S}_m / s = 0\}$ $Q \cup \{pos(s)\}$ $P = \left\{ \text{combinação } C_2^Q \right\}$ $p = (a, b) \text{ sendo}$ $a = ij, b = kl \text{ para}$ $i \leq k, j < l \text{ se } i = k$ <p>retorne $\mathbf{S}_{s'}$</p>
--

Figura 5.1: Movimento bl_1 .

$$\begin{array}{c}
 \mathbf{S} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & \textcircled{1} & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \\
 \oplus \\
 \mathbf{S}_m = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & \textcircled{1} & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \\
 \Downarrow \\
 \mathbf{S}' = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & \textcircled{0} & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}
 \end{array}$$

Figura 5.2: Movimento 1 - ação= bl_{1_a} .

5.3.3 Simulated Annealing (SA)

O método do *Simulated Annealing* busca melhorar a solução fornecida, procurando evitar regiões de atração de ótimos locais. A estratégia realiza movimentos com certo grau de aleatoriedade, visando ampliar a área de exploração do espaço de estados e encontrar soluções

Pseudo-código 14 Movimento 1 (S_{atual})

```

1:  escolhe ação por sorteio
2:   $S_{s'} \leftarrow bl_1(\text{ação}, S_{atual}, S_m)$ 
3:  seleciona o movimento de menor valor de custo para  $S' \in S_{s'}$ 
4:  Se  $S'$  factível então
5:    Se  $Custo(S') < Custo(S_{atual})$  então
6:       $S_{atual} \leftarrow S'$ 
7:    Fim Se
8:  Fim Se
Retorna:  $S_{atual}$ 

```

mais diversificadas. Alguns movimentos tratam o conhecimento prévio sobre o problema, outros realizam uma exploração ampla do espaço de estados e os demais procuram manter a generalidade da solução inicial.

O processo possui um estrutura semelhante à apresentada na BI, porém nesta estratégia dois tipos de movimentos são sorteados e executados a cada iteração. Esses dois tipos de movimentos são escolhidos através da elaboração de dois subconjuntos que contém movimentos diferentes. O primeiro subconjunto possui um tipo de movimento que se assemelha àqueles executados nas BIs e o segundo subconjuntos possui tipos de movimentos mais diversificados.

É sorteado um tipo de movimento para cada subconjunto, produzindo uma nova solução. O melhor movimento, dado pelo valor de custo da função objetivo, é selecionado para a continuação do procedimento.

Se o valor do custo dessa solução for menor que o valor do custo dado pela solução atual, a solução atual é substituída por ela. Senão, é executado o procedimento com uma probabilidade p .

Calcula-se a probabilidade p usando a Equação 4.2 da Seção 4.3. A solução atual é substituída pela solução encontrada com probabilidade p e o processo de busca no espaço continua. Senão, o parâmetro que mede o grau de funcionalidade do método SA analisa o número de ocorrências da solução sucessora dada pelos movimentos, determinando, ou não, o fim da busca.

Além desse critério de parada, outros dois critérios podem determinar a interrupção do processo. O primeiro é o parâmetro que mede a eficácia dos movimentos executados. Se nenhuma solução sucessora for encontrada pelos movimentos, a busca é encerrada. O segundo critério de parada é o número de iterações, onde a busca para, quando um número máximo de processos de transição executados for atingido. No término da busca, uma solução final é retornada. O Pseudo-código 15, ilustra esse processo de busca realizado pelo método *Simulated Annealing*.

Pseudo-código 15 Simulated Annealing (\mathbf{S}_{final})

```

1:  $CustoFinal \leftarrow Custo(\mathbf{S}_{final})$ 
2: inicia  $T$ 
3: Repita
4:   efetua movimento 1 sorteado
5:   efetua movimento 2 sorteado
6:   Guarda as soluções
7:   Seleciona a melhor delas e armazena em  $\mathbf{S}'_{final}$ 
8:   Se  $Custo(\mathbf{S}'_{final}) < Custo(\mathbf{S}_{final})$  então
9:      $\mathbf{S}_{final} \leftarrow \mathbf{S}'_{final}$ 
10:    Se  $Custo(\mathbf{S}'_{final}) < CustoFinal$  então
11:       $CustoFinal \leftarrow Custo(\mathbf{S}'_{final})$ 
12:    Fim Se
13:  Senão
14:    faça  $\mathbf{S}_{final} \leftarrow \mathbf{S}'_{final}$  com probabilidade  $p$ 
15:    atualiza  $T$ 
16:  Fim Se
17: Até que (Limite de Iteração ou Parâmetros de Parada Satisfeitos)
Retorna:  $\mathbf{S}_{final}$ 

```

Movimentos do SA

Os movimentos efetuados no SA são separados em dois subconjuntos de movimentos com características distintas. No primeiro subconjunto os movimentos conservam a mesma característica da solução inicial; no segundo, os movimentos são aleatórios, com e sem conhecimento prévio do problema.

- Movimento 2.1 - Mesma características da solução inicial.

Esse movimento tem estrutura de execução semelhante à realizada nas BIs, sendo diferente em dois aspectos. Não utiliza a solução modelo para determinar os pares que sofrerão a ação sorteada, pois neste movimento todos serão analisados. Porém, a execução da ação está mais restrita, pois só sofrerão ação os pares onde os dois elementos estiverem aptos ao movimento. O processo de transição de estados do Movimento bl_{21} é ilustrado na Fig.5.3. O Movimento bl_{21} , \mathbf{S} é a solução atual, \mathbf{S}_1 é uma solução modelo definida com todos elementos iguais a 0 (sem manutenção), $\mathcal{S}_{s'}$ é o conjunto que contém todas as soluções sucessoras \mathbf{S}' produzidas, pos é o operador que define a posição do elemento, os índices i e k representam a linha e os índices j e l representam a coluna da matriz descrita na Seção 4.2.

A avaliação da solução e o complemento do processo de transição de estados segue a mesma estratégia descrita no movimento anterior. Deste modo, o processo iterativo do Movimento 2.1 com essa avaliação da solução é ilustrado no Pseudo-código 16. A Fig.5.4 ilustra o processo de transição do movimento em um determinado par de elementos da solução atual.

$bl_{21}(\mathbf{S}, \text{ação})$ Se ação == bl_{21_a} então $\forall p_1 \in P_1$ $\mathbf{S}' \leftarrow \mathbf{S}$ $s'(a) = 1$ e $s'(b) = 0$ $\mathbf{S}_{s'} \cup \mathbf{S}'$ Senão Se ação == bl_{21_b} então $\forall p_2 \in P_2$ $\mathbf{S}' \leftarrow \mathbf{S}$ $s'(b) = 0$ e $s'(a) = 1$ $\mathbf{S}_{s'} \cup \mathbf{S}'$ P_1 e $P_2 \leftarrow seldepare(\mathbf{S}, \mathbf{S}_1)$ $\forall s \in \{\mathbf{S} \oplus \mathbf{S}_1 / s = 0\}$ $Q_1 \cup \{pos(s)\}$ $\forall s \in \{\mathbf{S} \oplus \mathbf{S}_1 / s = 1\}$ $Q_2 \cup \{pos(s)\}$ $\forall a \in Q_1$ e $\forall b \in Q_2$ $P_1 \cup \{(a, b)\}$ $P_2 \cup \{(b, a)\}$ sendo $a = ij, b = kl$ para $i \leq k, j < l$ se $i = k$ retorne $\mathbf{S}_{s'}$

Figura 5.3: Movimento bl_{21} .

$$\begin{aligned}
 \mathbf{S} &= \begin{bmatrix} 1\textcircled{0}0\textcircled{0}101010101011011010101010100 \\ 10110010101010101000000011000000 \\ 11101010000000110010101010101011 \end{bmatrix} \\
 &\quad \oplus \\
 \mathbf{S}_1 &= \begin{bmatrix} 0\textcircled{0}0\textcircled{0}000000000000000000000000 \\ 00000000000000000000000000000000 \\ 00000000000000000000000000000000 \end{bmatrix} \\
 &\quad \Downarrow \\
 \mathbf{S}' &= \begin{bmatrix} 1\textcircled{0}0\textcircled{0}0101010101011011010101010100 \\ 10110010101010101000000011000000 \\ 11101010000000110010101010101011 \end{bmatrix}
 \end{aligned}$$

Figura 5.4: Movimento 2.1 - ação= bl_{21_a} .

No segundo subconjunto, os movimentos podem ser separados em dois tipos diferentes de estratégia. Um tipo de movimento tem características mais amplas, operando em vários elementos da solução de uma vez. O outro possui características aleatórias, onde os elementos que sofrem a ação são determinados por instruções específicas dadas pelo conhecimento da

Pseudo-código 16 Movimento 2.1 (\mathbf{S}_{atual})

```

1:  seleciona ação
2:   $\mathbf{S}_{s'} \leftarrow bl_{21}(\mathbf{S}_{atual}, \text{ação})$ 
3:  seleciona o movimento de menor valor de custo para  $\mathbf{S}' \in \mathbf{S}_{s'}$ 
4:  Se  $\mathbf{S}'$  factível então
5:    Se  $Custo(\mathbf{S}') < Custo(\mathbf{S}_{atual})$  então
6:       $\mathbf{S}_{atual} \leftarrow \mathbf{S}'$ 
7:    Fim Se
8:  Fim Se
Retorna:  $\mathbf{S}_{atual}$ 

```

formulação do problema.

Dentre os tipos de movimentos mais amplos, figuram três tipos de estratégias, adotando a ideia de operar a ação em vários elementos da solução.

- Movimento 2.2.1.1 - Movimentos aplicados a elementos escolhidos a partir de uma Progressão Aritmética.

Este movimento calcula a progressão aritmética (PA) com razões $r = 2$ ou $r = HP$ (HP é o último estágio do horizonte de planejamento), escolhidas aleatoriamente. A cada vez que o movimento é sorteado, define-se a razão r e, seleciona-se uma parte da solução que vai sofrer o processo de transição. Na parte selecionada da solução \mathbf{S}' são operadas as ações de r em r elementos começando pelo primeiro. Essa ação executa a alteração do nível de manutenção de cada elemento, alternadamente e em toda solução. Por exemplo, se o nível de manutenção do primeiro elemento selecionado for aumentado, o segundo elemento terá seu nível de manutenção reduzido, seguindo até o último elemento possível de ser selecionado. Os elementos selecionados só sofrem a ação se estiverem aptos a isso. Esse procedimento fornece dois tipos de ações que podem ser operadas no movimento. Porém, somente um tipo é executado, sendo escolhido por sorteio no início do movimento.

Apenas uma solução sucessora é elaborada por esse tipo de movimento. Essa solução tem sua factibilidade testada, e caso essa seja factível, o valor do custo dado pela função objetivo é avaliado, junto ao valor de custo da solução atual. Se o valor de custo dessa nova solução for melhor, a solução atual é substituída por essa nova solução e a busca continua. O processo de transição de estados é ilustrado na Fig.5.5. Nesse movimento, \mathbf{S} é a solução atual e \mathbf{S}' é a solução final; r é a razão escolhida para a PA, os índices i e k representam a linha e os índices j e l representam a coluna da matriz descrita na Seção 4.2. O Pseudo-código 17 representa a iteração da transição, com o processo de avaliação, dado pelo movimento. A Fig.5.6 ilustra o processo de transição do movimento com razão igual a 2.

- Movimento 2.2.1.2 - Movimentos aplicados a elementos escolhidos aleatoriamente.

$bl_{2211}(r, ação, \mathbf{S})$ Seja $r = 2$ ou $r = HP$ Se $ação == bl_{2211_a}$ então Para $i = 1, \dots, HP$ e $j = 1, \dots, \lfloor \frac{n-1}{2r} \rfloor$ $P_1 \cup \{(i, 1 + 2rj)\}$ Para $k = 1, \dots, HP$ e $l = 1, \dots, \lfloor \frac{n-(1+r)}{2r} \rfloor$ $P_2 \cup \{(k, 1 + r + 2rl)\}$ $\forall p_1 \in P_1$ e $\forall p_2 \in P_2$ $s'_{p_1} \leftarrow 1$ e $s'_{p_2} \leftarrow 0$ Senão Se $ação == bl_{2211_b}$ então Para $i = 1, \dots, HP$ e $j = 1, \dots, \lfloor \frac{n-1}{2r} \rfloor$ $P_1 \cup \{(i, 1 + 2rj)\}$ Para $k = 1, \dots, HP$ e $l = 1, \dots, \lfloor \frac{n-(1+r)}{2r} \rfloor$ $P_2 \cup \{(k, 1 + r + 2rl)\}$ $\forall p_1 \in P_1$ e $\forall p_2 \in P_2$ $s'_{p_1} \leftarrow 0$ e $s'_{p_2} \leftarrow 1$ retorne \mathbf{S}'

Figura 5.5: Movimento bl_{2211} .

$$\begin{aligned}
 \mathbf{S} &= \begin{bmatrix} \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} \\ \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} \\ \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{1} \end{bmatrix} \\
 &\quad \downarrow \\
 \mathbf{S}' &= \begin{bmatrix} \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{1} & \textcircled{0} \\ \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} \\ \textcircled{1} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} & \textcircled{0} & \textcircled{0} & \textcircled{0} & \textcircled{1} \end{bmatrix}
 \end{aligned}$$

Figura 5.6: Movimento 2.2.1.1 - $ação=bl_{2211_a}$.

O objetivo desse movimento é aumentar a diversidade na transição de estados, ampliando a área do espaço de estados explorado. Escolhe-se para cada um dos elementos qual ação deverá ser operada, alterando apenas os elementos que estiverem aptos a recebê-la.

Nessa estratégia de movimento apenas uma solução sucessora é produzida. Testa-se a factibilidade dessa solução e, caso ela seja confirmada, o valor de custo é avaliado do mesmo modo que o movimento anterior. O processo de transição de estados é ilustrado na Fig.5.7. Nesse movimento, \mathbf{S} é a solução atual, \mathbf{S}' é a solução sucessora, o índice i representa a linha e o índice j representa a coluna da matriz descrita na Seção 4.2. Esse procedimento de transição, em conjunto com o processo de avaliação da solução sucessora, é desenvolvido no Pseudo-código 18. A Fig.5.8 ilustra o processo de transição do movimento operando em uma determinada solução atual.

Pseudo-código 17 Movimento 2.2.1.1 (\mathbf{S}_{atual})

```

1: Escolhe-se a razão  $r$  por sorteio
2: Escolhe-se a  $ação$  por sorteio
3:  $\mathbf{S}' \leftarrow bl_{2211}(r, ação, \mathbf{S}_{atual})$ 
4: Se  $\mathbf{S}'$  factível então
5:   Se  $Custo(\mathbf{S}') < Custo(\mathbf{S}_{atual})$  então
6:      $\mathbf{S}_{atual} \leftarrow \mathbf{S}'$ 
7:   Fim Se
8: Fim Se
Retorna:  $\mathbf{S}_{atual}$ 

```

$bl_{2212}(\mathbf{S})$ $\forall s_{ij} \in \mathbf{S}$ para $i = 1, \dots, HP$ e $j = 1, \dots, n$ Se $rand(0, 1] \leq 0.5$ então $s'_{ij} = 0$ Senão $s'_{ij} = 1$ retorne \mathbf{S}'

Figura 5.7: Movimento bl_{2212} .
$$\mathbf{S} = \begin{bmatrix} 1000100100100100100100100100100100100100100 \\ 10110001001001001001000000000110000000 \\ 11101010000000000110001010101010101011 \end{bmatrix}$$

↕

$$\mathbf{S}' = \begin{bmatrix} 1000100100110011001100110011001100011100111 \\ 01000010001000100001001000100010010101001 \\ 11001100100100000001100011100111001000 \end{bmatrix}$$

Figura 5.8: Movimento 2.2.1.2.

- Movimento 2.2.1.3 - Movimentos aplicados a elementos escolhidos um a um, testando o decréscimo do nível de manutenção.

O terceiro tipo de movimento 2.2.1.3 verifica se o nível de manutenção decresceu para cada elemento da solução. Cada elemento de cada ano do período que estiver apto a receber a ação operada será alterado a cada iteração do processo de transição. Várias soluções sucessoras são produzidas e guardadas no conjunto \mathbf{S}_s' , selecionando a solução \mathbf{S}' com menor valor de custo da função objetivo. A solução sucessora escolhida tem sua factibilidade testada, e o valor de custo é avaliado usando a função objetivo.

Esse tipo de movimento avalia se o nível de manutenção de um elemento da solução atual pode ser reduzido sem ultrapassar o limite de confiabilidade dado pelo valor FEC_{perm} , portanto melhorando o valor do custo da solução. O processo de transição de estados desse

Pseudo-código 18 Movimento 2.2.1.2 (\mathbf{S}_{atual})

```

1:  $\mathbf{S}' \leftarrow bl_{2212}(\mathbf{S}_{atual})$ 
2: Se  $\mathbf{S}'$  factível então
3:   Se  $Custo(\mathbf{S}') < Custo(\mathbf{S}_{atual})$  então
4:      $\mathbf{S}_{atual} \leftarrow \mathbf{S}'$ 
5:   Fim Se
6: Fim Se
Retorna:  $\mathbf{S}_{atual}$ 

```

movimento é ilustrado na Fig.5.9. Nesse movimento, \mathbf{S} é a solução atual, \mathbf{S}_2 é uma solução modelo definida com todos elementos iguais a 1 (com manutenção) e \oplus é o operador lógico *ou exclusivo*. Essa procedimento de transição de estados em conjunto com o processo de avaliação pode ser ilustrado no Pseudo-código 19. O processo de transição do movimento operando em um determinado elemento da solução atual é ilustrado na Fig.5.10.

$bl_{2213}(\mathbf{S}, \mathbf{S}_2)$ $\forall s' \in \{\mathbf{S} \oplus \mathbf{S}_2 / s' = 0\}$ $\mathbf{S}' \leftarrow \mathbf{S}$ $s' \leftarrow 0$ $\mathbf{S}_{s'} \cup \mathbf{S}'$ retorne $\mathbf{S}_{s'}$
--

Figura 5.9: Movimento bl_{2213} .

$$\begin{array}{c}
\mathbf{S} = \begin{bmatrix} \textcircled{1}0010101010101110110101010100 \\ 101100101010101000000011000000 \\ 111010100000001100101010101011 \end{bmatrix} \\
\oplus \\
\mathbf{S}_2 = \begin{bmatrix} \textcircled{1}1111111111111111111111111111 \\ 111111111111111111111111111111 \\ 111111111111111111111111111111 \end{bmatrix} \\
\Downarrow \\
\mathbf{S}' = \begin{bmatrix} \textcircled{0}001010101010101110110101010100 \\ 101100101010101000000011000000 \\ 111010100000001100101010101011 \end{bmatrix}
\end{array}$$

Figura 5.10: Movimento 2.2.1.3.

Seguindo com o segundo subconjunto de movimentos, têm-se definidos os outros tipos de estratégias de movimentos explorados. São estratégias com conhecimento *a priori* do problema. Características como a do parâmetro α_t , definido na Subseção 2.4.3, ou como a

Pseudo-código 19 Movimento 2.2.1.3 (\mathbf{S}_{atual})

```

1:  $\mathbf{S}_{s'} \leftarrow bl_{2213}(\mathbf{S}_{atual}, \mathbf{S}_2)$ 
2: seleciona o movimento de menor custo para  $\mathbf{S}' \in \mathbf{S}_{s'}$ 
3: Se  $\mathbf{S}'$  factível então
4:   Se  $Custo(\mathbf{S}') < Custo(\mathbf{S}_{atual})$  então
5:      $\mathbf{S}_{atual} \leftarrow \mathbf{S}'$ 
6:   Fim Se
7: Fim Se
Retorna:  $\mathbf{S}_{atual}$ 

```

função CB , definida na Subseção 5.3.1, determinam a estratégia das transições de estados. Dentre elas figuram os seguintes tipos de movimentos.

- Movimento 2.2.2.1 - Movimentos com base no valor da qualidade de serviço.

Essa estratégia utiliza as características dada pelo valor FEC_{perm} . Dado um elemento da rede, a realização ou não de sua manutenção preventiva nos primeiros anos do horizonte de planejamento determina o valor da qualidade de serviço. Sendo assim, o objetivo dessa estratégia de movimento é testar os elementos quanto a realização da ação de manutenção preventiva em determinados períodos.

Duas versões podem ser analisadas nessa estratégia de movimento. A primeira busca diminuir o valor do FEC no primeiro ano, realizando manutenção preventiva na maioria dos equipamentos desse período. A segunda busca diminuir o valor do custo de manutenção realizada no primeiro ano, realizando o mínimo de manutenção preventiva nesse período.

Essas duas versões exploram características distintas presentes na formulação do problema. Pelo fato da qualidade de serviço ser acumulativa ao longo dos anos, a primeira versão da estratégia executada é importante quando a confiabilidade é baixa. A segunda versão é importante quando esse limite escolhido pela restrição da qualidade de serviço é maior, explorando assim apenas a característica do parâmetro α_t para reduzir o valor de custo da função objetivo.

Na primeira versão, todos os equipamentos são analisados, escolhendo-se pares (a,b) de equipamentos, sendo $a = ij$, para $i = 1, \dots, HP-1$ e $j = 1, \dots, n$ e $b = kl$, para $k = i+1, \dots, HP$ e $l = 1, \dots, n$. A ideia da estratégia é fazer com que o par de equipamentos selecionado tenha a característica de realizar a manutenção preventiva no primeiro equipamento (a), e não realizar manutenção preventiva no segundo equipamento (b). Esse procedimento é realizado independente da existência ou não da manutenção prevista para o equipamento. Na segunda versão o procedimento é semelhante, porém a estratégia é inversa, não realiza a manutenção preventiva no primeiro equipamento (a), e realiza manutenção preventiva no segundo equipamento (b).

Nas duas versões, várias soluções sucessoras são elaboradas. Após o movimento executado, escolhe-se a solução que fornece o menor valor de custo, testando, a sua factibilidade. Se for

factível, o seu valor de custo calculado pela função objetivo é avaliado, e comparado ao valor de custo da solução atual, determinando, ou não, se essa solução atual é substituída. O processo de transição de estados desse movimento é ilustrado na Fig.5.11. Nesse movimento, \mathbf{S} é a solução atual, $\mathbf{S}_{s'}$ é o conjunto que contém todas soluções sucessoras \mathbf{S}' , os índices i e k representam a linha e os índices j e l representam a coluna da matriz descrita na Seção 4.2. A estratégia do processo de transição de estados do movimento em conjunto com a avaliação da solução sucessora, pode ser visualizada no Pseudo-código 20. O processo de transição do movimento operando em um determinado par da solução atual é ilustrado na Fig.5.12.

$bl_{2221}(\mathbf{S}, ação)$
Se ação == bl_{2221_a} então $\forall s_{ij} \in \mathbf{S}$ para $i = 1, \dots, (HP - 1)$ e $j = 1, \dots, n$ e $\forall s_{kl} \in \mathbf{S}$ para $k = 1 + i, \dots, (HP), l = 1, \dots, n$ $P \cup \{ij, kl\}$ $\forall p(a, b) \in P$ $\mathbf{S}' \leftarrow \mathbf{S}$ $s'_a \leftarrow 1$ $s'_b \leftarrow 0$ $\mathbf{S}_{s'} \cup \{\mathbf{S}'\}$ Senão Se ação == bl_{2221_b} então $\forall s_{ij} \in \mathbf{S}$ para $i = 1, \dots, (HP - 1)$ e $j = 1, \dots, n$ e $\forall s_{kl} \in \mathbf{S}$ para $k = 1 + i, \dots, (HP), l = 1, \dots, n$ $P \cup \{ij, kl\}$ $\forall p(a, b) \in P$ $\mathbf{S}' \leftarrow \mathbf{S}$ $s'_a \leftarrow 0$ $s'_b \leftarrow 1$ $\mathbf{S}_{s'} \cup \{\mathbf{S}'\}$ retorne $\mathbf{S}_{s'}$

Figura 5.11: Movimento bl_{2221} .

- Movimento 2.2.2.2 - Movimentos através da medida de investimento.

Esse movimento gera uma outra solução modelo através da função CB , para ser utilizada como objeto de comparação com a solução atual. Essa nova solução modelo é semelhante à solução modelo utilizada na BI, sendo diferente em apenas um aspecto. Enquanto a solução modelo da BI é dada a partir da média das medidas de investimentos dadas pela função CB , a nova solução modelo dependerá dessa média e de um parâmetro que define o seu crescimento,

$$\begin{array}{c}
\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \\
\Downarrow \\
\mathbf{S}' = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}
\end{array}$$

Figura 5.12: Movimento 2.2.2.1 - ação= bl_{2221a} .

Pseudo-código 20 Movimento 2.2.2.1 (\mathbf{S}_{atual})

- 1: escolhe-se a *ação* por sorteio
 - 2: $\mathbf{S}_{s'} \leftarrow bl_{2221}(\mathbf{S}_{atual}, ação)$
 - 3: seleciona o movimento de menor custo para $\mathbf{S}' \in \mathbf{S}_{s'}$
 - 4: **Se** \mathbf{S}' *factível* **então**
 - 5: **Se** $Custo(\mathbf{S}') < Custo(\mathbf{S}_{atual})$ **então**
 - 6: $\mathbf{S}_{atual} \leftarrow \mathbf{S}'$
 - 7: **Fim Se**
 - 8: **Fim Se**
- Retorna:** \mathbf{S}_{atual}
-

cada vez que esse movimento é executado. O parâmetro *controledesolução*, é uma variável crescente e tem seu valor modificado à cada vez que esse tipo de estratégia for sorteado.

O parâmetro é calculado e multiplicado pela média, gerando a nova solução da mesma forma que a solução modelo da BI foi criada. Cada elemento de um determinado ano do período da solução atual é comparado com o elemento correspondente na nova solução modelo. Se os elementos forem diferentes, o elemento da solução atual é armazenado separadamente em dois subconjuntos, conforme seu nível de manutenção.

Combina-se dois a dois os elementos dos subconjuntos, sendo selecionado um de cada subconjunto. Todas as combinações possíveis que tiverem aptas a sofrerem alterações no seu nível de manutenção passarão pelo processo de transição de estados do movimento. Cada par de elementos produz uma solução sucessora \mathbf{S}' , que será armazenada em um conjunto $\mathbf{S}_{s'}$. Terminado o processo de transição, escolhe-se a solução sucessora que fornece o melhor movimento (menor custo de função objetivo). Se essa solução escolhida for factível, ela passa pelo procedimento de avaliação do valor do custo dada pela função objetivo, determinando, ou não, a substituição da solução atual. Esse processo de transição de estados é ilustrado na Fig.5.13. Nesse movimento, \mathbf{S} é a solução atual, $\mathbf{S}_{s'}$ é o conjunto que contém todas soluções sucessoras \mathbf{S}' , *pos* é o operador que define a posição do elemento e \oplus é o operador lógico *ou exclusivo*. O procedimento de transição de estados combinado com o processo de avaliação é resumido no Pseudo-código 21. A Fig.5.14 ilustra o procedimento de transição do movimento

em um determinado par de elementos da solução atual.

$bl_{2222}(\mathbf{S}, \mathbf{S}_m)$
$\mathbf{S}' \leftarrow \{\mathbf{S} \oplus \mathbf{S}_m\}$
$\forall s' \in \{\mathbf{S}' / s' = 0\}$
$P \cup \{pos(s)\}$
$\forall p \in P$
Se $s_p = 0$ então
$P_1 \cup \{p\}$
Senão
$P_2 \cup \{p\}$
$\forall p_1 \in P_1$
$\mathbf{S}'' \leftarrow \mathbf{S}$
$s''_{p_1} \leftarrow 1$
$\mathbf{S}_{s''} \cup \{\mathbf{S}''\}$
$\forall p_2 \in P_2$
$\mathbf{S}'' \leftarrow \mathbf{S}$
$s''_{p_2} \leftarrow 0$
$\mathbf{S}_{s''} \cup \{\mathbf{S}''\}$
retorne $\mathbf{S}_{s''}$

Figura 5.13: Movimento bl_{2222} .

$$\begin{aligned}
 \mathbf{S} &= \begin{bmatrix} \textcircled{1}0\textcircled{0}1010101010111011010101010100 \\ 10110010101010101000000011000000 \\ 11101010000000110010101010101011 \end{bmatrix} \\
 &\quad \oplus \\
 \mathbf{S}_m &= \begin{bmatrix} \textcircled{1}1\textcircled{0}10101101010100111111111111 \\ 110101001101010101001100101010 \\ 0000001001000000010000000111 \end{bmatrix} \\
 &\quad \Downarrow \\
 \mathbf{S}' &= \begin{bmatrix} \textcircled{0}0\textcircled{1}1010101010111011010101010100 \\ 10110010101010101000000011000000 \\ 11101010000000110010101010101011 \end{bmatrix}
 \end{aligned}$$

Figura 5.14: Movimento 2.2.2.2.

Uma característica importante desse movimento é como o cálculo do parâmetro *controle-desolução* influencia a criação da solução modelo. O parâmetro é multiplicado pela média das medidas de investimento dos elementos e, como o parâmetro é crescente, a média também será. Assim, elaborada a solução modelo, o número de elementos sem manutenção aumentará, o que ajuda a busca pela redução dos valores de custo.

Pseudo-código 21 Movimento 2.2.2.2 (\mathbf{S}_{atual})

- 1: Calcula-se o parâmetro *controledesolução*
 - 2: Cria-se \mathbf{S}_m (*controledesolução*)
 - 3: $\mathcal{S}_{s''} \leftarrow bl_{2222}(\mathbf{S}, \mathbf{S}_m)$
 - 4: seleciona o movimento com o menor custo para $\mathbf{S}'' \in \mathcal{S}_{s''}$
 - 5: **Se** \mathbf{S}'' *factível* **então**
 - 6: **Se** $Custo(\mathbf{S}'') < Custo(\mathbf{S}_{atual})$ **então**
 - 7: $\mathbf{S}_{atual} \leftarrow \mathbf{S}''$
 - 8: **Fim Se**
 - 9: **Fim Se**
- Retorna:** \mathbf{S}_{atual}
-

5.3.4 Fluxogramas

Para representar a junção desses processos de iteração é apresentado um fluxograma para cada metodologia discutida, ilustrados nas Figuras Fig. 5.15, Fig. 5.16 e Fig. 5.17.

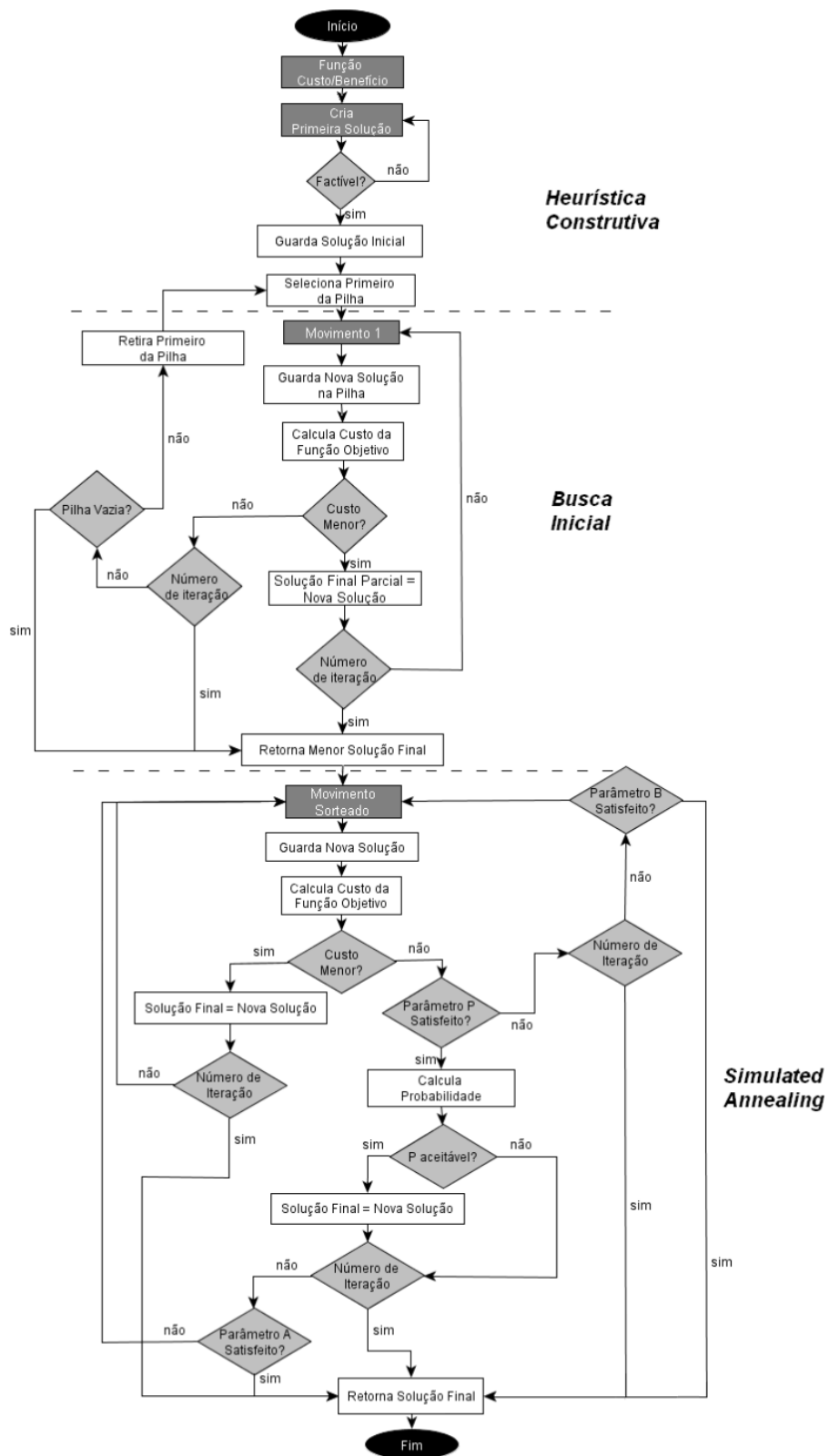


Figura 5.15: Fluxograma - Metodologia 1.

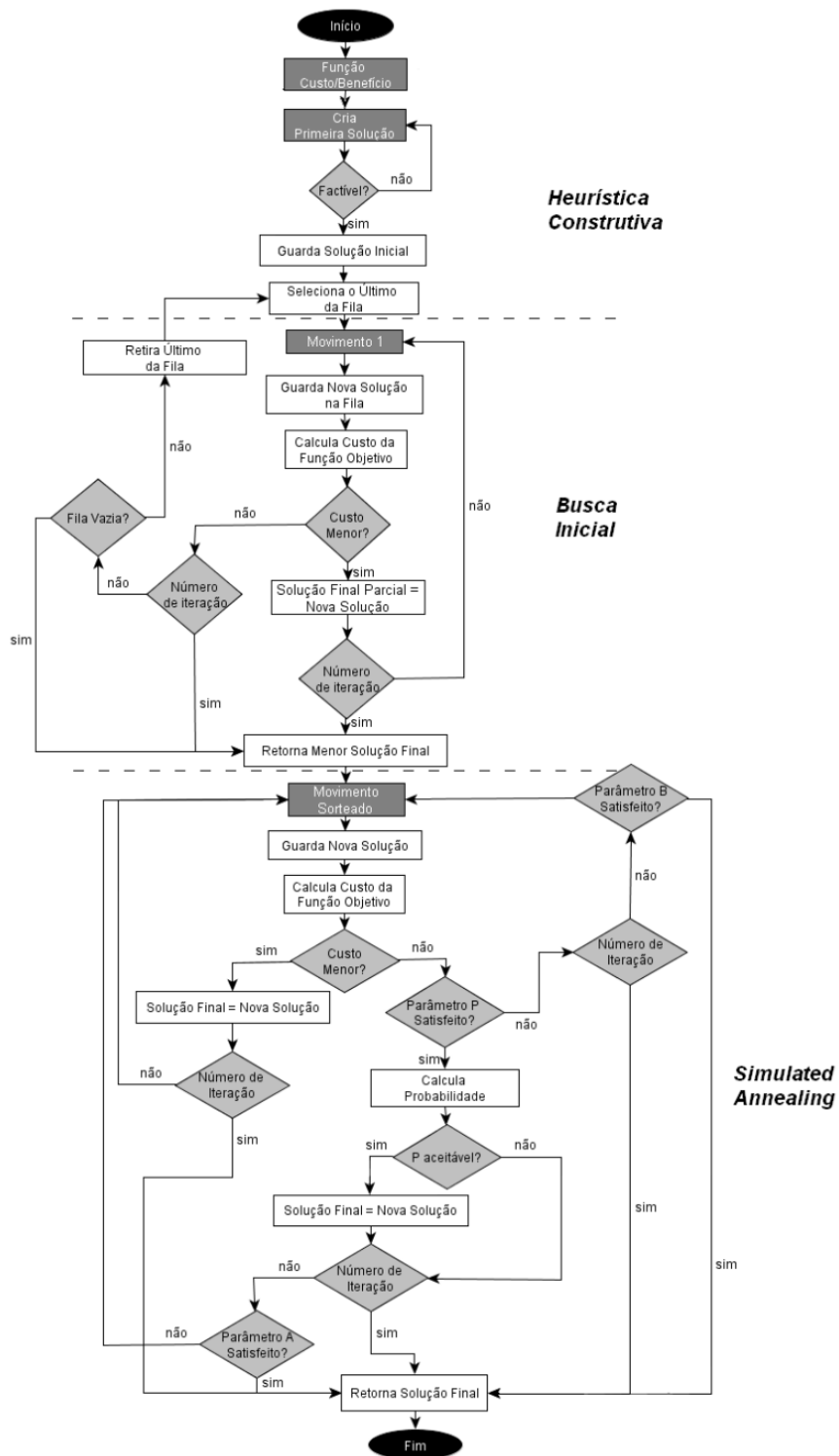


Figura 5.16: Fluxograma - Metodologia 2.

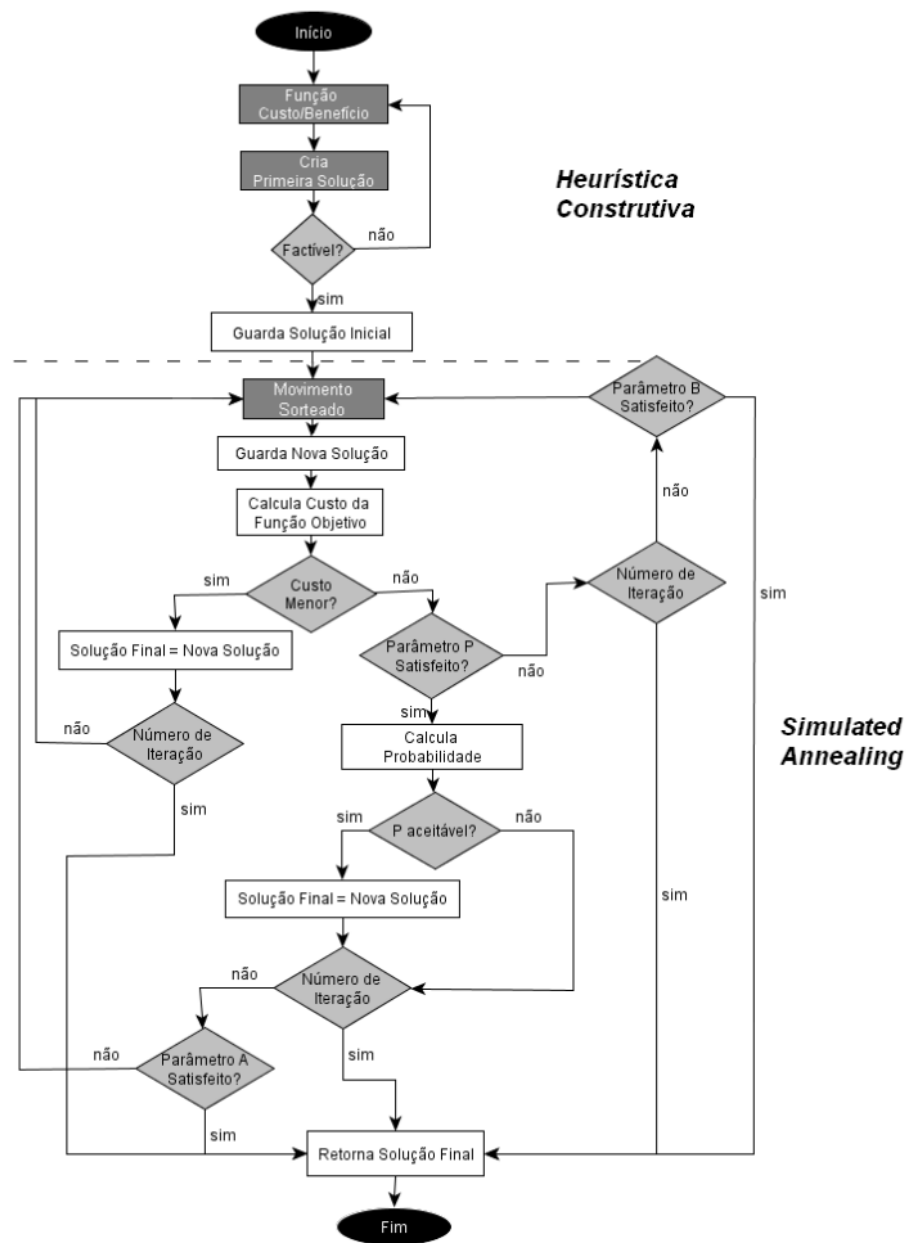


Figura 5.17: Fluxograma - Metodologia 3.

Estudo dos Casos

6.1 Introdução

Os estudos de casos foram elaborados utilizando a rede de referência apresentada no trabalho de Reis (2007).

É estudada também uma ampliação da rede de referência, com o objetivo de testar a eficiência das metodologias desenvolvidas para redes de maior porte.

Alguns experimentos são elaborados, visando testar a robustez da metodologia e a adaptação do algoritmo à situações adversas de restrição e de período. Cada experimento apresentará resultados obtidos, produzindo uma comparação entre as metodologias desenvolvidas e a metodologia abordada por Reis (2007).

6.2 Rede de Referência

Devido a dificuldade de encontrar dados suficientes na literatura para o problema de manutenção em SDEE, a rede estudada como referência nesta dissertação é uma rede radial desenvolvida em *Sittithumwat et. al.* (2004) e adaptada por Reis (2007). Contudo, para representar diferentes tipos de experimentos foi necessária uma adaptação a essa Rede de Referência. Ela foi expandida, definindo novos equipamentos para as seções definidas, porém com os mesmos valores de custo de manutenção preventiva, custo de manutenção corretiva, taxas de falhas iniciais e multiplicadores de taxa de falha de cada tipo de equipamento utilizados. O diagrama da Rede de Referência é apresentado na Fig.6.1.

Cada bloco ilustrado na Fig.6.1 define uma seção, que por sua vez é determinada pelos equipamentos de segurança de um sistema, tais como, os religadores, transformadores e reguladores.

Os blocos mais escuros definem as seções formadas por redes de distribuição primária e os mais claros definem as seções formadas por redes de distribuição secundária. O número

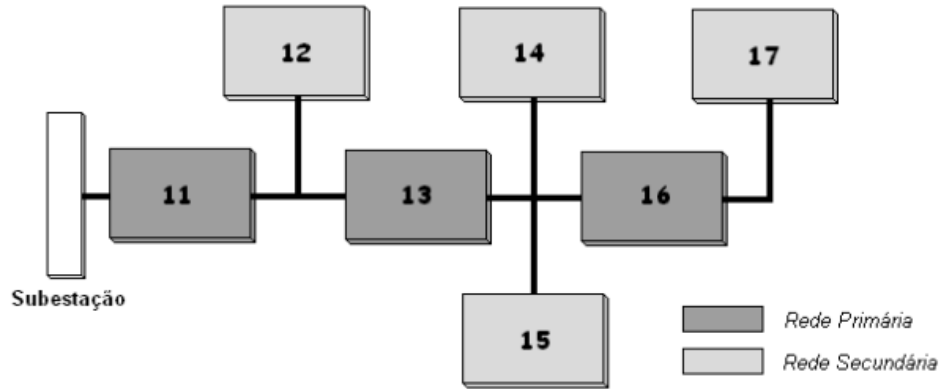


Figura 6.1: Rede de Referência

interno ao bloco indica uma seção, sendo 11 a seção mais próxima da subestação de energia justificando, a sua importância na rede e o número de clientes que ela abrange.

6.2.1 Casos Explorados

Dois sistemas de rede de distribuição são testados no caso da Rede de Referência. O primeiro caso é a rede de referência que contém 30 equipamentos distribuídos em sete seções. O segundo caso é uma ampliação da rede contendo 76 equipamentos distribuídos nas mesmas sete seções. Nos dois casos existem dez tipos diferentes de equipamentos apresentados na subseção seguinte.

As Tabelas 6.1 e 6.2 ilustram a distribuição dos equipamentos na rede representada na Fig.6.1 apresentada na seção anterior. Nas tabelas a coluna Equipamento representa o número que identifica cada equipamento na rede. Na coluna Seção são apresentadas as seções onde o equipamento está localizado. As Seções 11, 12, 13, 14, 15, 16 e 17 são tratadas como equipamentos (poda de rede primária e secundária) e por isso terão valores de custos e taxas de falhas definidos. Além disso, na coluna seção para esses equipamentos é informada a estrutura da rede. Por exemplo, a Seção 11 está ligada à (subestação), Seção 12 está ligada à Seção 11, e assim por diante, como está representado na Fig.6.1.

6.2.2 Dados da Rede e Equipamentos

Os dados da rede se resumem ao número de clientes que uma seção abrange e o número total de clientes. Os dados dos equipamentos são compostos por valores de custo de manutenção preventiva e corretiva, valores de taxas iniciais de falha, os multiplicadores de taxas de falha e as taxas de falha iniciais.

O número de clientes atendidos por seção e o número total de clientes mudam conforme

Equipamento	Seção	Tipo
11	-1	Poda1
12	11	Poda2
13	11	Poda1
14	13	Poda2
15	13	Poda2
16	13	Poda1
17	16	Poda2
21	11	Relig.
31	12	Transf.
32	14	Transf.
33	15	Transf.
34	17	Transf.
41	13	Regul.
51	11	Cabo
52	13	Cabo
53	16	Cabo
61	11	Poste1
62	13	Poste1
63	16	Poste1
71	12	Poste2
72	14	Poste2
73	15	Poste2
74	17	Poste2
81	11	Infra1
82	13	Infra1
83	16	Infra1
91	12	Infra2
92	14	Infra2
93	15	Infra2
94	17	Infra2

Tabela 6.1: Primeiro Caso de Sistema de Rede.

Eq.	Seção	Tipo	Eq.	Seção	Tipo	Eq.	Seção	Tipo
11	-1	Podal	121	11	Relig.	221	11	Relig.
12	11	Podal	131	12	Transf.	231	12	Transf.
13	11	Podal	132	14	Transf.	232	14	Transf.
14	13	Podal	133	15	Transf.	233	15	Transf.
15	13	Podal	134	17	Transf.	234	17	Transf.
16	13	Podal	141	13	Regul.	241	13	Regul.
17	16	Podal	151	11	Cabo	251	11	Cabo
21	11	Relig.	152	13	Cabo	252	13	Cabo
31	12	Transf.	153	16	Cabo	253	16	Cabo
32	14	Transf.	161	11	Poste1	261	11	Poste1
33	15	Transf.	162	13	Poste1	262	13	Poste1
34	17	Transf.	163	16	Poste1	263	16	Poste1
41	13	Regul.	171	12	Poste2	271	12	Poste2
51	11	Cabo	172	14	Poste2	272	14	Poste2
52	13	Cabo	173	15	Poste2	273	15	Poste2
53	16	Cabo	174	17	Poste2	274	17	Poste2
61	11	Poste1	181	11	Infra1	281	11	Infra1
62	13	Poste1	182	13	Infra1	282	13	Infra1
63	16	Poste1	183	16	Infra1	283	16	Infra1
71	12	Poste2	191	12	Infra2	291	12	Infra2
72	14	Poste2	192	14	Infra2	292	14	Infra2
73	15	Poste2	193	15	Infra2	293	15	Infra2
74	17	Poste2	194	17	Infra2	294	17	Infra2
81	11	Infra1						
82	13	Infra1						
83	16	Infra1						
91	12	Infra2						
92	14	Infra2						
93	15	Infra2						
94	17	Infra2						

Tabela 6.2: Segundo Caso de Sistema de Rede.

a rede analisada, porém o número de tipos de equipamentos apresentados e suas respectivas características não são alterados.

Como foi citado na subseção anterior, dez tipos de equipamentos estão presentes na estrutura das redes. Seus valores de custo de manutenção, multiplicadores de taxa de falha e taxas de falha iniciais são apresentados na Tabela 6.3.

Tipo	CMC	CMP	Mtx MP	CSM	Mtx SM	TF Inicial
Cabo	0.06	0.03	0.92	0	1.08	0.02
Infraestrutura 1	0.94	0.47	0.79	0	1.26	0.05
Infraestrutura 2	0.94	0.47	0.79	0	1.26	0.05
Poste 1	14.5	7.25	0.69	0	1.2	0.001
Poste 2	14.5	7.25	0.69	0	1.2	0.001
Regulador	16	8	0.89	0	1.12	0.029
Religador	1.2	0.6	0.91	0	1.28	0.015
Poda de Rede 1	2.05	1.025	0.95	0	1.51	0.05
Poda de Rede 2	1.05	0.525	0.95	0	1.51	0.05
Transformador	1.692	0.846	0.95	0	1.51	0.01

Tabela 6.3: Tipos de Equipamentos.

Na Tabela 6.3 as siglas *CMC*, *CMP*, *Mtx MP*, *CSM*, *Mtx SM* e *TF Inicial* representam respectivamente os valores de Custo de Manutenção Corretiva, Custo de Manutenção Preventiva, Multiplicador da Taxa de Falhas para a ação de Manutenção Preventiva, Custo Sem Manutenção, Multiplicador da Taxa de Falhas para a ação Sem Manutenção e Taxa de Falhas Iniciais.

Conforme já foi relatado, o número de clientes que cada seção e o número total de clientes varia de acordo com a rede executada. Essa mudança é refletida através do aumento do número de equipamento de um caso para o outro. Esse fato é detalhado na Tabela 6.4.

	Rede 30 Equipamentos	Rede 76 Equipamentos
	Nº de Clientes	Nº de Clientes
Seção 11	5000	15000
Seção 12	1100	3300
Seção 13	3700	11100
Seção 14	1100	3300
Seção 15	1100	3300
Seção 16	1300	3900
Seção 17	1100	3300
Total	5000	15000

Tabela 6.4: Número de Clientes por Seção e Número Total de Clientes em uma Rede.

É importante ressaltar a distribuição dos valores para cada seção, sendo intuito notar que as seções de redes de distribuição primária possuem um número maior de clientes do que as

de rede de distribuição secundária. Esse fato é explicado na Seção 2.4.3 do Capítulo 2, onde a importância de uma seção está ligada diretamente à proximidade da subestação. Quanto mais próxima uma seção está da subestação e quanto mais seções forem dependentes de sua funcionalidade, maior sua importância e o número de clientes relacionados.

6.3 Experimentos

Os experimentos a serem realizados visam testar a robustez dos algoritmos desenvolvidos, através da mudança de alguns aspectos da formulação do problema: Mudança do período para o planejamento de manutenção (Horizonte de Planejamento); mudança da taxa de juros aplicada (α_t); e mudança do limite imposto pela restrição da qualidade de serviço (FEC). Todas esses aspectos determinam mudanças na solução final obtida pelo modelo.

Primeiramente, as metodologias são testadas a fim de justificar sua escolha e estrutura, além de exemplificar os métodos apresentados no Capítulo 4.

As metodologias são analisadas através do valor da função objetivo dada pela solução final encontrada e do tempo de processamento, destacando posteriormente a análise comparativa das metodologias desenvolvidas nesse trabalho com a abordagem por algoritmo genético desenvolvida em (Reis 2007).

6.3.1 Experimento I - Variação do FEC

O primeiro experimento realizado busca testar o algoritmo quanto a variação da restrição do problema (FEC). Para a representação do FEC_{perm} para o problema são escolhidos três cenários:

- A restrição mínima dada pela rede executada;
- A restrição máxima dada pela rede executada;
- Uma restrição próxima da média desses valores.

O cálculo da restrição mínima da rede é obtido quando se tem a melhor qualidade de serviço, ou seja, quando todos os equipamentos estão sujeitos à ação de manutenção preventiva. A restrição mínima será um valor bem próximo a esse cálculo, para que o algoritmo encontre a melhor solução dentro de um cenário de rigidez na regulamentação. O cálculo da restrição máxima da rede é obtido quando se tem a pior qualidade de serviço, ou seja, quando todos os equipamentos estão sujeitos à ação sem manutenção preventiva. A restrição máxima será o próprio valor obtido, testando o algoritmo para casos sem nenhuma restrição na regulamentação. Os valores de FEC, são calculados de acordo com a Equação 2.1 da Seção 2.4.3, apresentados na Tabela 6.5.

	Caso I - 30 Equipamentos	Caso II - 76 Equipamentos
FEC mínimo	0,3305	0,7178
FEC máximo	0,9311	1,8018

Tabela 6.5: FEC máximo e mínimo.

Calculados o FEC máximo e mínimo, pode-se determinar os valores de FEC adotados para o Experimento I. Para o Caso da Rede I os valores de FEC são 0,35, 0,55 e 0,95 e para o Caso da Rede II são 0,72, 1,00 e 1,82.

Cada metodologia foi executada cinco vezes, sendo que o valor de custo escolhido é o menor encontrado e o tempo de processamento é a média dos tempos de execução. Além disso, é importante relatar que o valor de Horizonte de Planejamento (HP) é igual para todas as metodologias testadas ($HP = 3$), variando apenas a taxa de juros anual.

Na Tabela 6.6 são apresentados os valores do custo e do tempo médio de processamento obtidos através da variação do FEC para o Caso de Rede I. As tabelas variam o FEC e o valor da taxa de juros anual.

Para melhor ilustrar a diferença entre os valores finais de custo encontrados pelos algoritmos é construído o gráfico de variação dos custos conforme as mudanças executadas e apresentado na Fig.6.2.

Na Tabela 6.7 são apresentados os valores de custo e tempo médio de processamento para o Caso de Rede II obtidos, variando o valor do FEC e da taxa de juros anual.

Um gráfico de variação dos custos é elaborado e apresentado na Fig.6.3 para ilustrar a diferença entre os valores finais de custo encontrados pelos algoritmos.

A partir dessas tabelas e figuras produzidas, pode-se realizar uma análise dos algoritmos.

6.3.2 Análise do Experimento I

Os resultados obtidos são melhores no algoritmo BPS executado, tanto no valor do custo quanto no tempo de processamento. O algoritmo BLS também produz bons resultados, mostrando a importância da execução das BIs como um primeiro processo de busca. Esses resultados podem ser explicados pelo fato dos métodos executados na BIs direcionarem a solução, determinando uma melhor solução inicial para o segundo processo de busca. Com uma melhor solução inicial, a estratégia de *Simulated Annealing* economizará passos no processamento, melhorando seu desempenho computacional e a exploração no espaço de estados.

Um outro fato importante nessa análise, são os valores referentes à variação da taxa de investimento. Com a restrição de serviço mais rigorosa, a diferença entre os valores de cada metodologia, para cada taxa de investimento escolhida, mantém-se estável, porém quando a restrição vai deixando de ser rigorosa, a diferença entre os valores deixa de ser estável. No Caso de Rede I, quando a taxa de investimento aumenta o algoritmo BLS melhora seus

<i>FEC = 0,35 e Taxa de Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	24,6275	24,6324	24,7019
Tempo	8,39	10,57	9,92

<i>FEC = 0,35 e Taxa de Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	23,6826	23,7044	23,7121
Tempo	8,62	10,70	5,70

<i>FEC = 0,55 e Taxa de Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	14,2355	14,2871	14,3657
Tempo	7,16	7,66	6,24

<i>FEC = 0,55 e Taxa de Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	13,4375	13,4375	13,5640
Tempo	9,80	7,88	9,08

<i>FEC = 0,95 e Taxa de Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	07,2746	07,2746	07,2746
Tempo	5,23	0,58	0,42

<i>FEC = 0,95 e Taxa de Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	06,9220	06,9220	06,9220
Tempo	4,75	0,59	0,49

Tabela 6.6: Resultados Caso Rede I.

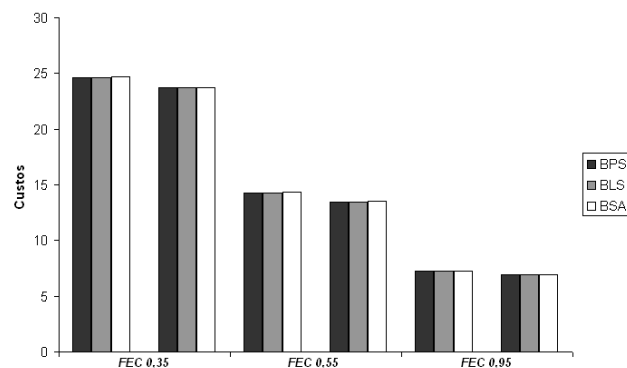


Figura 6.2: Custos Comparados - Caso I (HP=3).

<i>FEC = 0,72 e Taxa de Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	119,6642	120,5653	130,2311
Tempo	3075,65	6860,65	3643,27

<i>FEC = 0,72 e Taxa de Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	119,5900	121,3165	129,8979
Tempo	1978,92	7445,23	1707,42

<i>FEC = 1,00 e Taxa de Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	32,1219	32,3646	32,3402
Tempo	761,42	1742,24	2935,69

<i>FEC = 1,00 e Taxa de Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	30,4572	30,4662	30,4572
Tempo	377,42	797,24	903,22

<i>FEC = 1,82 e Taxa de Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	15,1706	15,1706	15,1706
Tempo	16,69	18,66	0,58

<i>FEC = 1,82 e Taxa de Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	14,4568	14,4568	14,4568
Tempo	16,69	9,78	0,57

Tabela 6.7: Resultados Caso Rede II.

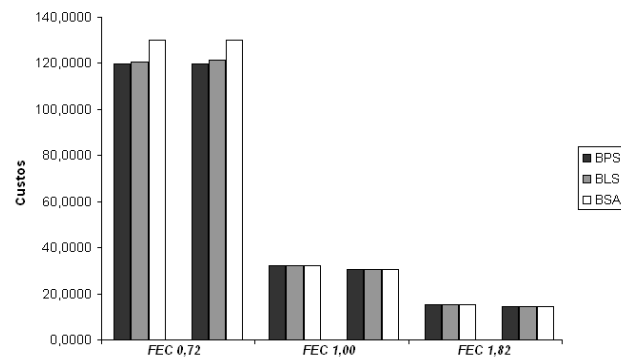


Figura 6.3: Custos Comparados - Caso II (HP=3).

resultados comparados com o algoritmo BPS, porém com tempo de processamento um pouco menor. Enquanto que os resultados encontrados pelo BSA apenas pioram.

No Caso de Rede II, os resultados obtidos pelo algoritmo BLS, quando se aumenta o valor da taxa de investimento, não apresentam melhorias. Contudo, os resultados obtidos pelo BSA apresentam melhoras significativas, sendo semelhante aos resultados obtidos pelo algoritmo BPS e melhores que os resultados obtidos pelo algoritmo BLS, mas com tempos de processamento maiores.

Comparando os resultados do Caso de Rede I com o Caso de Rede II, observa-se que o comportamento do algoritmo BPS é estável em relação ao número de equipamentos da rede.

O ponto fraco do algoritmo BPS é o tempo de processamento quando a restrição deixa de existir. Esses altos tempos de processamento podem ser explicados pelo fato das condições de parada desse algoritmo serem menos rigorosas do que as condições de paradas presentes nos outros algoritmos.

No Caso I, por se tratar de uma rede com um baixo número de equipamentos, os algoritmos possuem valores de tempo médio semelhantes, exceto o caso onde não existe restrição.

No Caso II, por se tratar de uma rede que possui um número maior de equipamentos, dois cenários para análise são produzidos. No primeiro cenário, tem-se a restrição de serviço mais rigorosa, onde se pode ver o alto tempo médio de processamento do algoritmo BLS. Como são poucas as soluções disponíveis no espaços de estados, várias soluções produzidas pelo processo de transição serão semelhantes. Assim, dada a característica do algoritmo BLS de explorar primeiro todos os sucessores, como foi mencionado na Subseção 4.2.2, muitas soluções semelhantes serão inseridas na fila a cada iteração, determinando o aumento do tempo necessário para o processamento. No segundo cenário, tem-se uma restrição menos rigorosa, onde o BSA apresenta os piores valores de tempos médios para o processamento. Como número de soluções factíveis no espaço de estados aumenta, mais diversificadas serão as soluções produzidas pelo processo de transição. Assim, como o BSA não possui uma boa solução inicial, ele precisará de mais iterações para encontrar uma boa solução final.

Nas Figs. 6.4 e 6.5 são apresentados os *boxplots*¹ referentes as cinco execuções efetuadas.

Pode-se notar que os *boxplots* presentes nas figuras não apresentam semelhanças quanto a sua estrutura, dado a mudança de caso de rede analisada. A diferença mais notável é a melhora do comportamento do algoritmo BLS, porém sem produzir melhoras no valor de custo final. O BPS foi o algoritmo que menos apresentou diferenças nos comportamentos, possuindo sempre o menor mínimo e o menor valor de 1º *quartil*. O BSA, apresentou a maior variação de valores de custos produzidos, determinando um comportamento ruim em ambos os casos de rede.

¹Gráfico que possibilita representar a distribuição de um conjunto de dados com base em alguns de seus parâmetros descritivos.

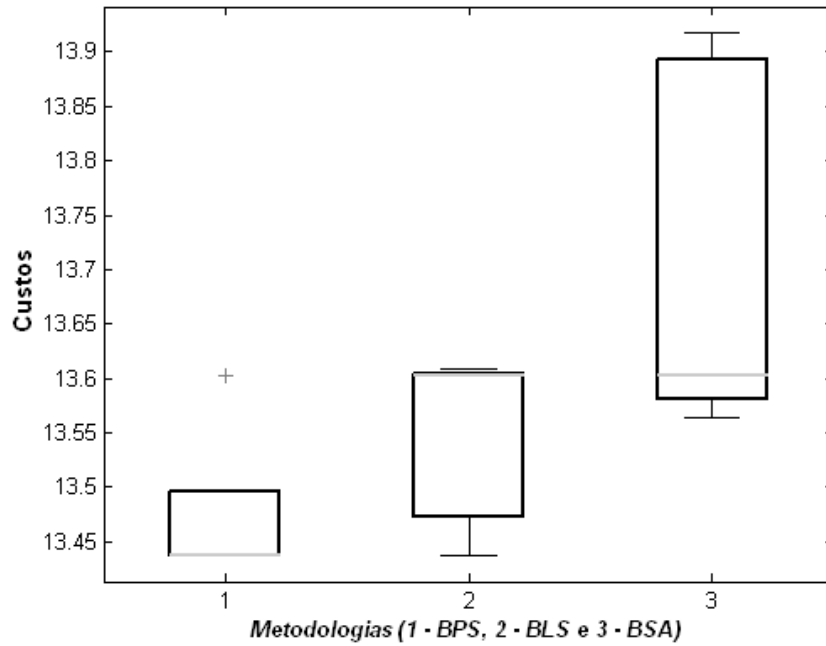


Figura 6.4: Boxplot 1 - Caso I - FEC = 0,55, HP=3 e Juros 15%.

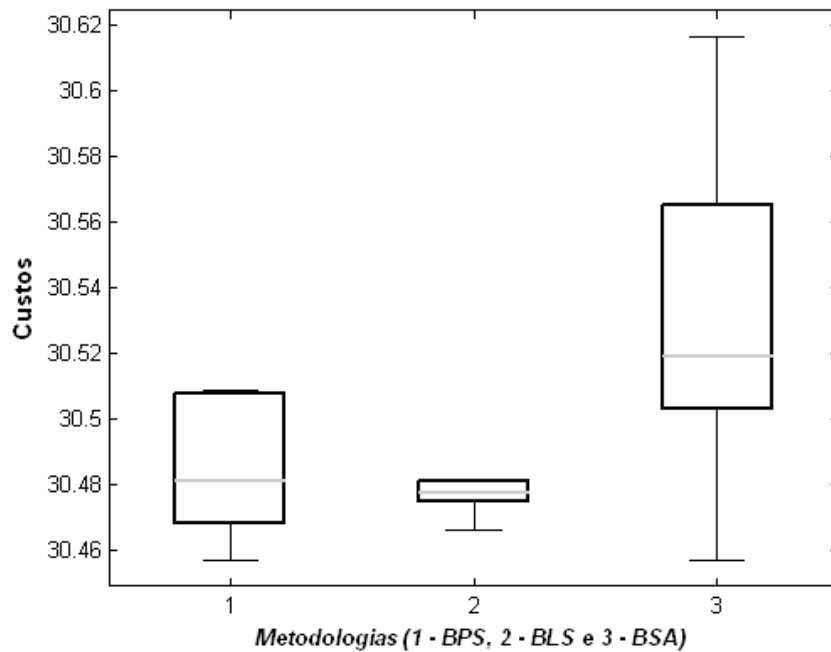


Figura 6.5: Boxplot 2 - Caso II - FEC = 1,00, HP=3 e Juros 15%.

6.3.3 Experimento II - Horizonte de Planejamento

Esse segundo experimento verifica o comportamento do algoritmo quanto a mudança do horizonte de planejamento do problema. Dois cenários são analisados para o Caso de Rede I:

- Caso I com $HP = 3$;
- Caso I com $HP = 5$.

O Caso de Rede II não será executado nesse experimento, pois os tempos de processamento para o caso com $HP = 5$ são muito altos, devido às limitações da linguagem de programação utilizada quando o número de elementos é muito grande.

Para a elaboração dos testes, as metodologias são executadas variando o FEC e a taxa de juros anual. Os valores de FEC utilizados são escolhidos com base no máximo, no mínimo e em um valor representativo, como foi citado no desenvolvimento do experimento anterior.

	Caso I - HP = 3	Caso I - HP = 5
FEC mínimo	0,3305	0,3305
FEC máximo	0,9311	1,8121

Tabela 6.8: FEC máximo e mínimo do Caso I.

Os valores definidos para o FEC são 0,35, 0,55 e 0,95 para $HP = 3$ e 0,35, 0,55 e 1,82 para $HP = 5$.

Cada metodologia foi executada cinco vezes, sendo que o valor de custo escolhido é o menor encontrado dentre os cinco e o tempo de processamento é a média dos tempos de execução. Nesse experimento, apenas o Caso I é executado, variando o valor atribuído à taxa de juros anual e a restrição de qualidade do problema.

São apresentados na Tabela 6.9 os valores obtidos pelas cinco execuções de cada metodologia para o Caso I, referentes aos valores de custos e tempo médio de processamento, para facilitar o processo de comparação dos algoritmos.

Na Fig.6.6 é ilustrada a diferença dos valores de custos obtidos pelos algoritmos através das mudanças definidas.

6.3.4 Análise do Experimento II

Em todas as mudanças executadas o algoritmo BPS se mostrou mais eficiente no valor de solução final. Ainda, obteve bons resultados quanto aos tempos médios de execução dado o aumento do horizonte de planejamento, exceto quanto a ausência de restrição. Nesse caso, os valores médios de tempo são grandes, devido aos tipos diferentes de condições de parada para cada algoritmo, sendo menos rigorosos no algoritmo BPS.

<i>HP=3, FEC = 0,35 e Tx.Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	24,6275	24,6324	24,7019
Tempo	8,39	10,57	9,92

<i>HP=5, FEC = 0,35 e Tx.Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	41,0824	41,0824	42,4476
Tempo	146,21	286,42	161,68

<i>HP=3, FEC = 0,35 e Tx.Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	23,6826	23,7044	23,7121
Tempo	8,62	10,70	5,70

<i>HP=5, FEC = 0,35 e Tx.Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	38,4762	38,9877	38,9192
Tempo	172,42	185,47	220,18

<i>HP=3, FEC = 0,55 e Tx.Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	14,2355	14,2871	14,3657
Tempo	7,16	7,66	6,24

<i>HP=5, FEC = 0,55 e Tx.Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	29,6738	29,6738	29,9202
Tempo	108,54	209,42	180,84

<i>HP=3, FEC = 0,55 e Tx.Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	13,4375	13,4375	13,5640
Tempo	9,80	7,88	9,08

<i>HP=5, FEC = 0,55 e Tx.Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	26,7379	26,8260	26,7671
Tempo	134,67	175,47	153,89

<i>HP=3, FEC = 0,95 e Tx.Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	7,2746	7,2746	7,2746
Tempo	3,80	0,58	0,42

<i>HP=5, FEC = 1,82 e Tx.Juros 10%</i>			
Métodos	BPS	BLS	BSA
Custo	15,5581	15,5581	15,5581
Tempo	34,11	3,24	0,65

<i>HP=3, FEC = 0,95 e Tx.Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	6,9220	6,9220	6,9220
Tempo	4,75	0,59	0,49

<i>HP=5, FEC = 1,82 e Tx.Juros 15%</i>			
Métodos	BPS	BLS	BSA
Custo	13,9953	13,9953	13,9953
Tempo	35,69	4,05	0,94

Tabela 6.9: Resultados HP = 3 e HP = 5.

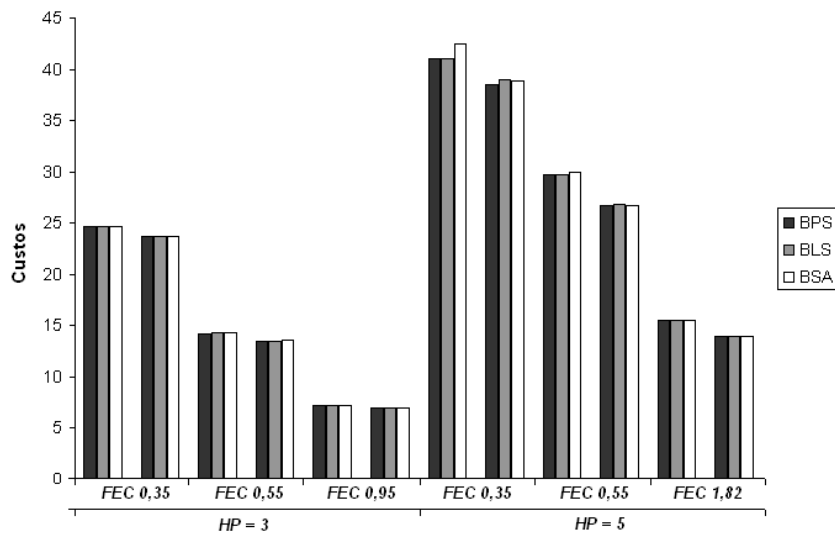


Figura 6.6: Custos Comparados - Caso I (HP=3 e HP=5).

O algoritmo BLS também encontrou bons resultados quanto ao valor de custo, sendo superado pelo algoritmo BSA em dois casos. Em ambos os casos, a rede estudada era maior ($HP=5$) e a taxa de investimento estava em seu maior patamar. Essa análise mostra o bom funcionamento do método *Simulated Annealing* em relação ao aumento da penalidade no valor do investimento, possuindo ainda melhores valores médios de tempo de processamento. Na Fig.6.7 e na Fig.6.8 são apresentados os *boxplots* do comportamento dos algoritmos para as situações simuladas. Nos *boxplots*, apenas o algoritmo BLS apresenta alguma semelhança na estrutura. Os algoritmos BPS e BSA são bem diferentes, porém apresentam um melhor comportamento em relação aos valores obtidos no *boxplot* da Fig. 6.7.

O algoritmo BPS mais uma vez apresentou os menores valores de mínimo e de 1º *quartil*. Além disso, nesses casos apresentados, também obteve as menores médias de valores.

Um aspecto a ser destacado nessa análise é o comportamento do algoritmo BSA no primeiro *boxplot*. As soluções obtidas foram muito parecidas, porém nenhuma forneceu um valor significativo, sugerindo que o método *Simulated Annealing* tenha ficado preso diante das características da solução fornecida pela heurística construtiva. Esse fato mostra a importância da aplicação das BIs como um primeiro processo na busca por melhores soluções.

O principal resultado dessa proposta de experimento foi mostrar a eficiência do algoritmo BPS. Dado o aumento do HP, o algoritmo BPS continua encontrando boas soluções com valores de custos baixos, além de apresentar os menores tempos médio de processamento.

6.4 Estudo de Comparação das Metodologias

O objetivo desse experimento é determinar através de uma análise de comparação qual metodologia possui melhor rendimento para o planejamento de manutenção.

Nesse experimento, o Algoritmo Genético Híbrido (AGH) desenvolvido em (Reis 2007) é utilizado como objeto de estudo para a análise de comparação dos valores de custo obtidos pelas metodologias desenvolvidas nesse trabalho. O AGH é executado 5 vezes para cada cenário estudado e seus resultados analisados com base na formulação do problema desenvolvida na Seção 2.4.3, selecionando o menor valor de custo dado pela função objetivo.

O tempo médio de processamento não entra no desenvolvimento do estudo desse experimento, pois o algoritmo AGH é desenvolvido em uma linguagem de programação (*JAVA*) e os algoritmos BPS, BLS e BSA em outra (*Matlab*). Assim, uma análise de comparação de tempo de execução seria incompatível.

Para o planejamento serão estudados os dois Casos de Rede desenvolvidos. No Caso I os algoritmos são executados variando o Horizonte de Planejamento ($HP = 3$ e $HP = 5$), o FEC (0,40, 0,55 e 0,95 para $HP = 3$ e 0,40, 0,55 e 1,82 para $HP = 5$) e a taxa de juros anual (10% e 15%).

No Caso II apenas o FEC (0,72, 1,00 e 1,82) e a taxa de juros anual (10% e 15%) variam,

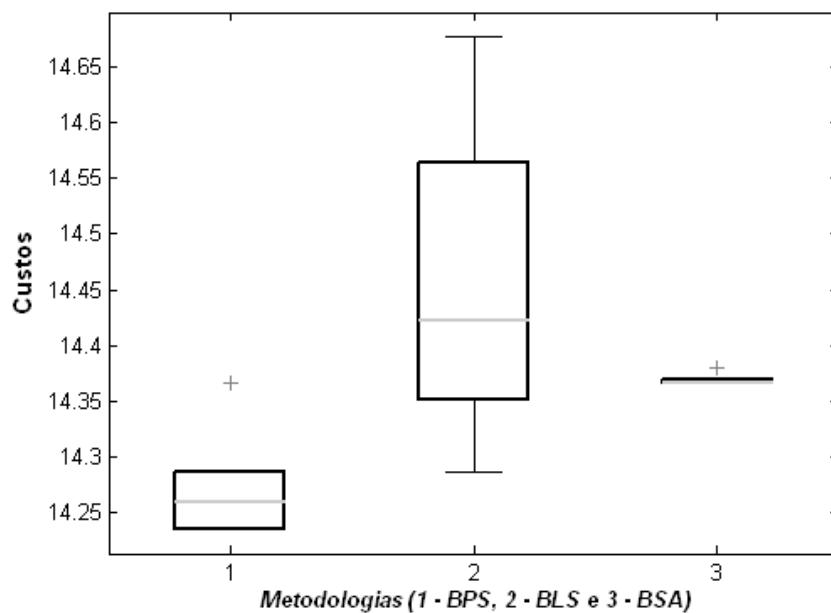


Figura 6.7: Boxplot 3 - Caso I - FEC = 0,55, HP=3 e Juros 10%.

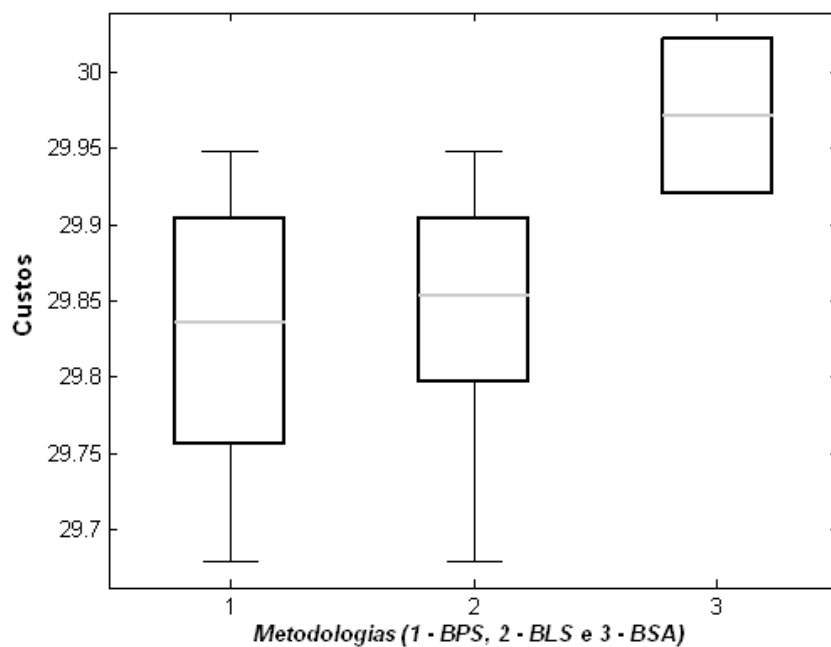


Figura 6.8: Boxplot 4 - Caso I - FEC = 0,55, HP=5 e Juros 10%.

mantendo o Horizonte de Planejamento fixo ($HP = 3$). Neste caso, o $HP=5$ não foi simulado pois o tempo de processamento foi muito alto.

Os resultados obtidos pelas cinco execuções são apresentados na Tabela 6.10, variando a cada linha da tabela os cenários propostos e a cada coluna o algoritmo executado.

Visando auxiliar a elaboração de uma análise detalhada do experimento realizados, são construídos três gráficos, apresentados na Fig.6.9, Fig.6.10 e Fig.6.11, que explicitam a variação dos custos por algoritmo executado.

6.4.1 Análise da Comparação

Quando a qualidade de serviço tem uma restrição mais rigorosa o AGH obtém boas soluções finais, sendo um pouco superiores que o algoritmo BPS. Contudo, quando o número de equipamentos da rede cresce, a diferença entre os resultados obtidos por esses algoritmos diminui, onde em alguns cenários o algoritmo BPS obtém melhores resultados. Para ilustrar essa análise, tem-se a Fig.6.9 e a Fig.6.10.

À medida que essa restrição de serviço vai deixando de ser rigorosa, os algoritmos BPS, BLS e BSA obtém bons resultados comparados ao AGH, superando-o em alguns cenários desenvolvidos. Quando a restrição do problema deixa de existir, todos os algoritmos testados obtém os mesmos resultados, mostrando que a estratégia de busca presente é eficiente quanto à liberdade de manutenção.

Os melhores resultados obtidos pelos algoritmos BPS, BLS e BSA são detalhados nos *boxplots* apresentados nas Figuras 6.12, 6.13 e 6.14. Os *boxplots* tem o objetivo de conduzir um outro tipo de análise sobre os algoritmos, para descrever o comportamento de cada um deles.

Com estruturas parecidas, figuram apenas o AGH e o algoritmo BPS, nos dois primeiros *boxplots*. Quanto ao aumento do número de equipamentos todos os algoritmos tiveram bom comportamento, obtendo sempre soluções com valores similares. Dentre essas soluções, o algoritmo BPS foi o que encontrou os menores valores de mínimo. Além disso, também encontrou os menores valores de média e os menores valores para o 1º *quartil*. O algoritmo BLS também obteve bons resultados, sendo bem semelhante aos obtidos pelo BPS. O algoritmo BSA, também obteve bons resultados, porém mostrou um desempenho ruim quando o número de equipamentos da rede é menor e a taxa de investimento também é pequena, determinando um dos piores comportamentos. Por fim, o AGH não obteve bons resultados comparados aos resultados obtidos pelos algoritmos desenvolvidos, apresentando sempre os piores resultados.

O fato do algoritmo BSA, no Caso de Rede I, apresentar comportamento diferente quanto a mudança da taxa de investimento, implica que o método de *Simulated Annealing* possui ações de transição de estados com características específicas para o valor de penalidade de

	AGH	BPS	BLS	BSA
Caso I				
HP=3, FEC=0,40 e Juros 10%	21,0907	21,1583	21,1560	21,1583
HP=3, FEC=0,40 e Juros 15%	20,0911	20,2307	20,2372	20,2307
HP=3, FEC=0,55 e Juros 10%	14,4249	14,2601	14,2871	14,3657
HP=3, FEC=0,55 e Juros 15%	13,6644	13,4375	13,4375	13,5640
HP=3, FEC=0,95 e Juros 10%	7,2746	7,2746	7,2746	7,2746
HP=3, FEC=0,95 e Juros 15%	6,9220	6,9220	6,9220	6,9220
HP=5, FEC=0,40 e Juros 10%	36,9316	37,5483	37,7355	37,6296
HP=5, FEC=0,40 e Juros 15%	33,4602	33,8446	34,6180	34,6495
HP=5, FEC=0,55 e Juros 10%	30,0111	29,6738	29,6738	29,9202
HP=5, FEC=0,55 e Juros 15%	27,4911	26,7379	26,8260	26,7671
HP=5, FEC=1,82 e Juros 10%	15,5581	15,5581	15,5581	15,5581
HP=5, FEC=1,82 e Juros 15%	13,9953	13,9953	13,9953	13,9953
Caso II				
HP=3, FEC=0,72 e Juros 10%	119,9894	119,6642	120,5653	130,2311
HP=3, FEC=0,72 e Juros 15%	118,1152	119,5900	121,3165	129,8979
HP=3, FEC=1,00 e Juros 10%	32,3090	32,1219	32,3646	32,3402
HP=3, FEC=1,00 e Juros 15%	30,8709	30,4572	30,4662	30,4572
HP=3, FEC=1,82 e Juros 10%	15,1706	15,1706	15,1706	15,1706
HP=3, FEC=1,82 e Juros 15%	14,4568	14,4568	14,4568	14,4568

Tabela 6.10: Comparação dos Valores de Custos.

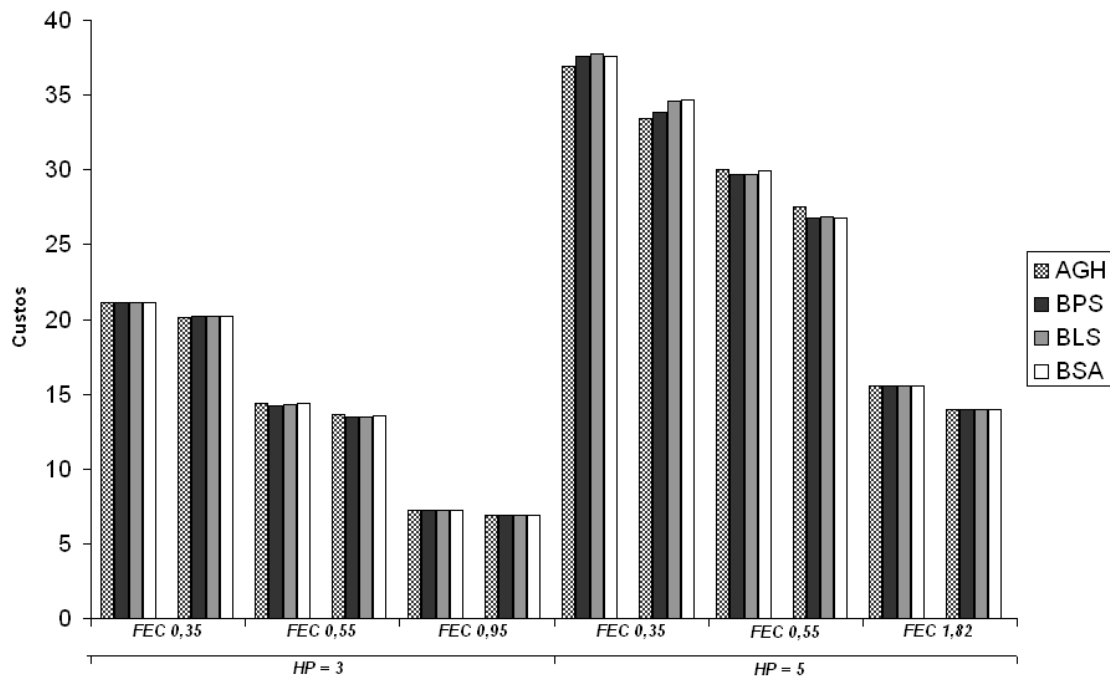


Figura 6.9: Valores de Custos - Caso I.

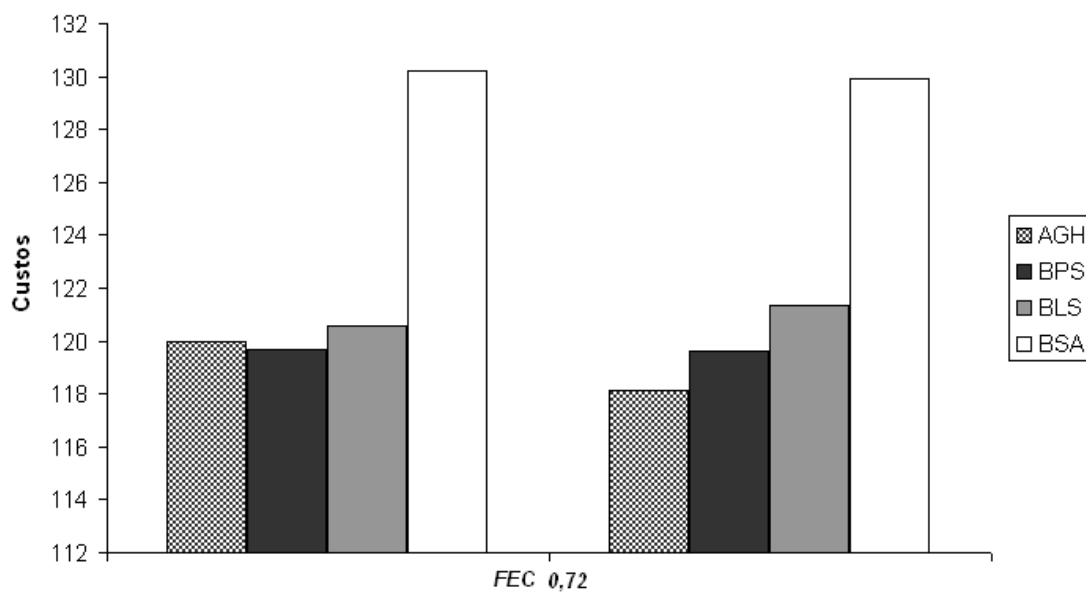


Figura 6.10: Valores de Custos - Caso II - FEC 0,72.

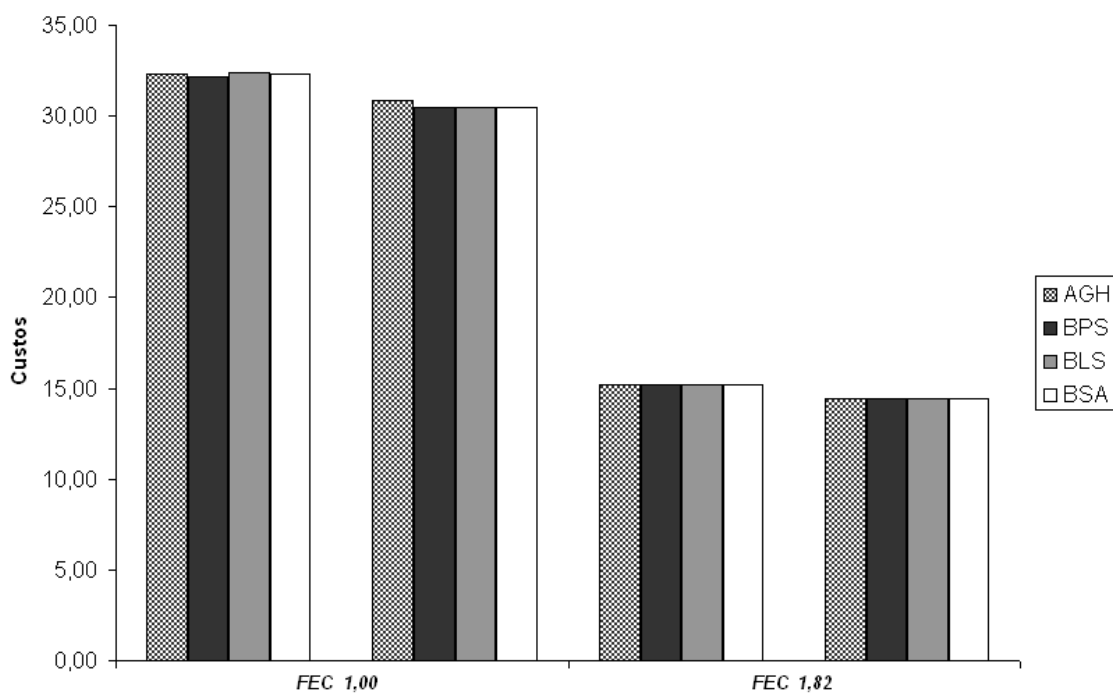


Figura 6.11: Valores de Custos - Caso II - FEC 1,00 e FEC 1,82.

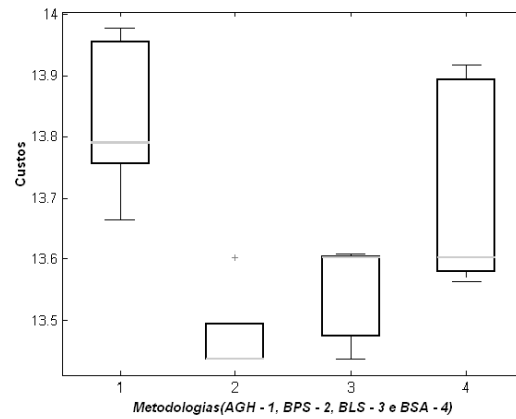


Figura 6.12: Boxplot 5 - Caso I - FEC = 0,55, HP=3 e Juros 15%.

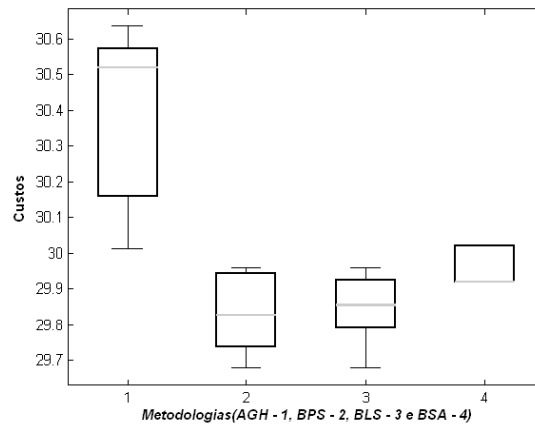


Figura 6.13: Boxplot 6 - Caso I - FEC = 0,55, HP=5 e Juros 10%.

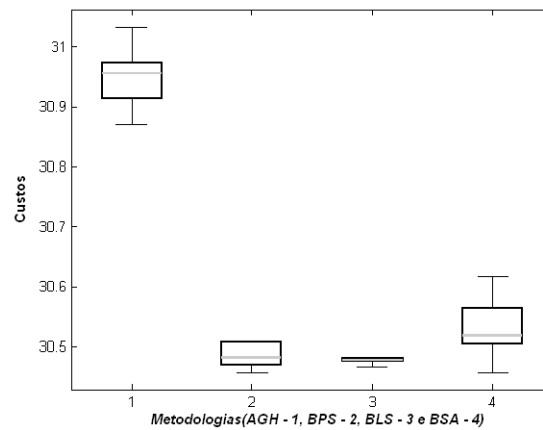


Figura 6.14: Boxplot 7 - Caso II - FEC = 1,00, HP=3 e Juros 15%.

terminado (α_t).

6.5 Comentários sobre a Análise da Comparação

Dada as soluções obtidas pelos algoritmos AGH e BPS para o Caso I com $FEC = 0,55$, $HP = 3$ e Taxa de Juros de 10%, pode-se observar através de suas representações, uma justificativa para o ganho obtido pelo algoritmo BPS.

AGH	=	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0									
		1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0								
		1	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0							
BPS	=	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
		1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0		
		1	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0

Figura 6.15: Soluções AGH e BPS.

Como foi descrito nos Capítulos 4 e 5, nas representações de soluções definidas, 0 significa a ação *sem manutenção preventiva* e 1 significa a ação *com manutenção preventiva*. Os exemplos de solução mostrados na Fig. 6.15, o AGH fornece um valor de custo de 14,6239 e o BPS um valor de custo de 14,2355. Essas soluções são representadas nas Figs. 6.16 e 6.17, que ilustram os equipamentos do sistema de rede estudado que sofrem manutenção preventiva.

Nota-se que, a solução obtida pelo algoritmo BPS possui mais equipamentos que sofrem manutenção preventiva, mesmo assim obtém um menor valor de custo (os equipamentos em destaque mostram essa afirmação). A Tabela 6.2 e os tipos de equipamentos na Tabela 6.3, justificam esses valores apresentados, pois se pode observar que os equipamentos nos quais o algoritmo BPS efetuou manutenção têm custos menores que os equipamento onde a AGH efetuou manutenção. Contudo, essas escolhas modificadas não alteram o cálculo da restrição de serviço, pois a diferença de valor do multiplicador associado a cada equipamento é compensada. Além disso, a disposição das manutenções preventivas efetuadas estão distribuídas de forma diferente privilegiando outros tipos de equipamentos.

AGH (Caso I - $FEC = 0.55$, $HP = 3$ e Taxa de investimento = 10%)
 Custo = 14,6239

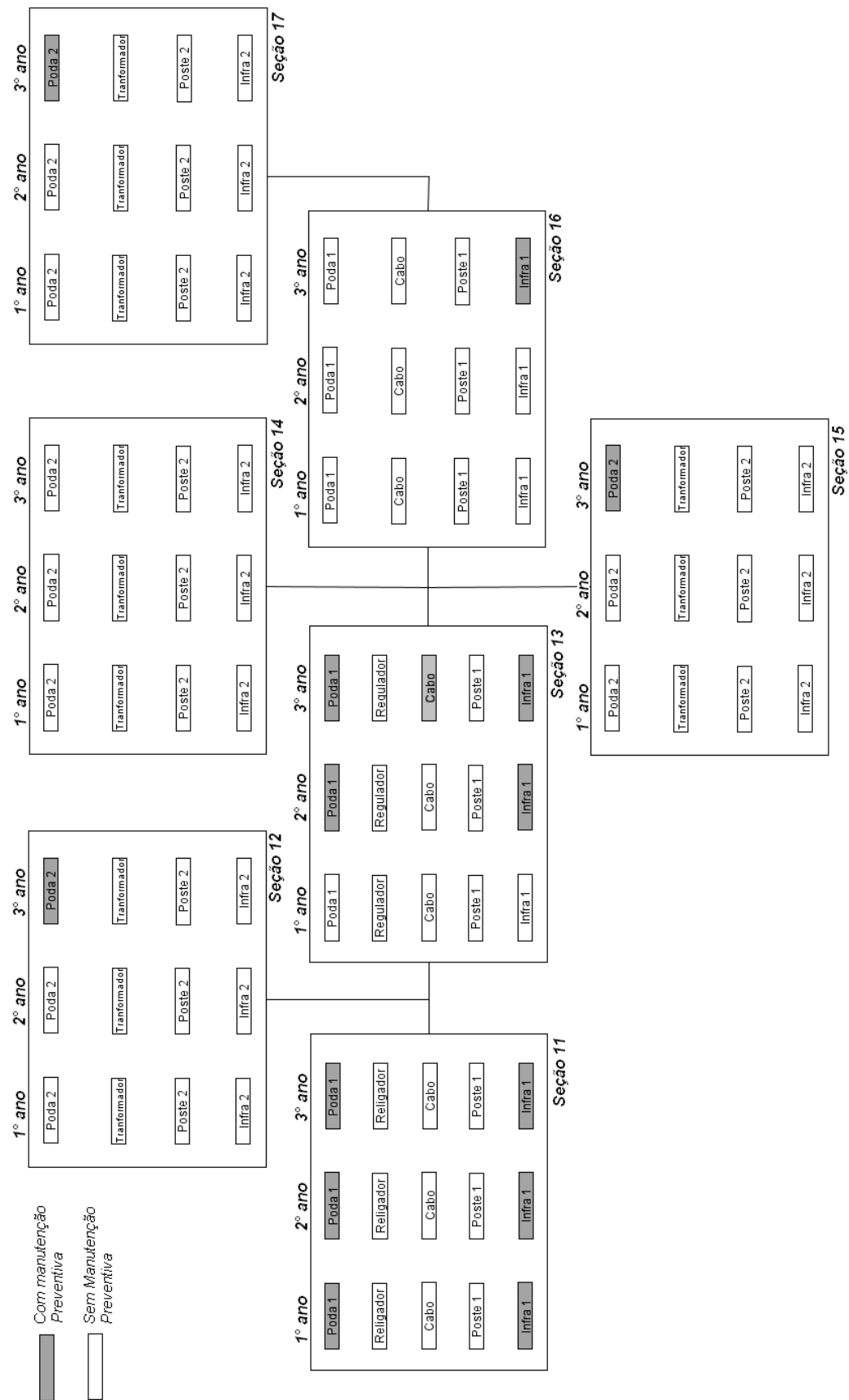


Figura 6.16: Solução - AGH.

BPS (Caso I - FEC = 0.55, HP = 3 e Taxa de investimento = 10%)
Custo = 14,2355

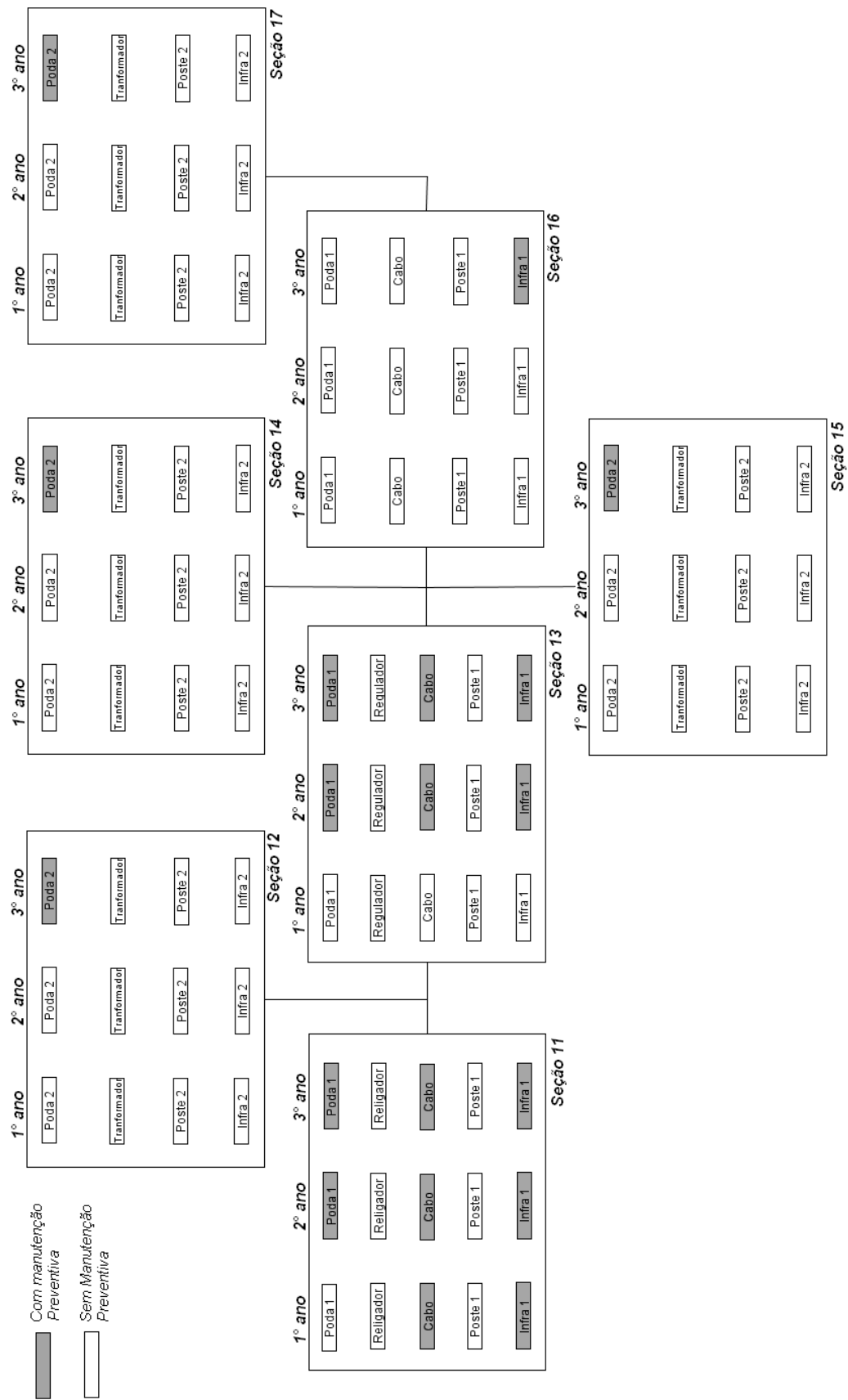


Figura 6.17: Solução - BPS.

Conclusões

A principal contribuição do trabalho foi analisar e desenvolver metodologias para o problema de alocação ótima de recursos para manutenção em redes de distribuição de energia elétrica. Sua principal referência foi o trabalho realizado por Reis (2007), que desenvolveu um modelo matemático para representar o problema e propôs uma abordagem para a solução, um algoritmo genético híbrido.

Este trabalho procura desenvolver alternativas de solução para o problema proposto. O principal subsídio metodológico para o desenvolvimento das heurísticas propostas no trabalho são os métodos de buscas em espaço de soluções candidatas, utilizados principalmente na área de inteligência artificial.

Foram desenvolvidas três heurísticas de buscas em espaços de estados para a abordagem do problema. Criou-se um conjunto de ações de transição de estados específicos para o problema estudado, que incorporam conhecimento sobre o problema e procedimentos de diversificação, com o objetivo de alcançar regiões promissoras do espaço de soluções, mas ao mesmo tempo procurando evitar, no espaço de busca, regiões onde soluções ótimas locais exercem atração excessiva.

A primeira heurística desenvolvida, denominada BPS, utiliza um procedimento construtivo guiado por uma função de avaliação do custo/benefício, seguido de uma *busca em profundidade* informada. Ao final da *busca em profundidade* é realizado um procedimento de melhoria baseado no método *Simulated Annealing*.

A segunda heurística desenvolvida, denominada BLS, utiliza um procedimento construtivo guiado por uma função de avaliação do custo/benefício, é seguido de *busca em largura* no espaços de estados, acrescida de informações sobre o problema para reduzir o espaço de busca. Ao final da *busca em largura*, é realizado um procedimento de melhoria baseado no método *Simulated Annealing*.

A terceira heurística desenvolvida, denominada BSA, realiza um procedimento construtivo guiado por uma função de avaliação do custo/benefício, seguido de uma versão do método

Simulated Annealing adaptada às características do problema.

Foram realizados dois conjuntos de estudos de casos. O primeiro conjunto compara entre si os procedimentos de otimização desenvolvidos neste trabalho, enquanto que o segundo conjunto compara os métodos desenvolvidos com o algoritmo genético híbrido proposto por Reis (2007).

Na análise dos resultados provenientes do primeiro conjunto de experimentos, o algoritmo BPS foi o que apresentou o melhor desempenho; obteve os melhores tempos de processamento na maioria dos estudos realizados e também obteve as soluções de melhor qualidade em todos os casos estudados.

As avaliações que comparam os métodos desenvolvidos com o algoritmo genético híbrido proposto por Reis (2007) permitem as seguintes observações:

- Nas redes de menor porte, os algoritmos de busca em espaço de estados forneceram as melhores soluções quando a restrição de confiabilidade foi menos rigorosa; em casos em que as restrições de confiabilidade foram mais fortes, o algoritmo genético híbrido apresentou os melhores resultados;
- Nas instâncias de maior porte, o algoritmo BPS obteve melhores resultados do que o algoritmo genético híbrido na maioria dos casos, tanto em situações com confiabilidade muito restrita, quanto em situações menos exigentes em termos de confiabilidade.

Dessa forma, pode-se concluir que a meta-heurística BPS, desenvolvida neste trabalho, apresentou um desempenho melhor que o algoritmo genético híbrido. Acredita-se que os processos de busca em espaços de estados obtiveram um melhor comportamento para esse tipo de problema por beneficiarem certos aspectos qualitativos desejáveis em políticas de manutenção. Alguns desses aspectos foram incorporados aos operadores de transição desenvolvidos neste trabalho. No entanto, há ainda um campo largo para investigações nessa direção. Por exemplo, se existirem restrições de investimentos anuais, pode-se dar preferência a alternativas de manutenção distribuídas no horizonte de planejamento.

Trabalhos futuros sob o mesmo paradigma de referência podem considerar outras alternativas de buscas informadas em espaços de estados. Em particular, parece promissor investigar as possibilidades de algoritmos inspirados no método de busca A^* . Outro possível estudo é a implementação dessas meta-heurísticas desenvolvidas como uma busca local para o AGH.

Os estudos de casos exploraram a elaboração de tabelas, gráficos e *boxplots* para a comparação das metodologias. Porém, outros recursos podem ser investigados futuramente, como mostra o artigo de Dolan & Moré (2002), que desenvolve um estudo sobre comparação de *softwares* científicos utilizando perfis de desempenho.

Bibliografia

- Aarts, E. & Korst, J. (1989). *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing*, John Wiley & Sons Ltd., Chichester.
- Atmar, W. (1994). Notes on the simulation of evolution, *IEEE Transactions on Neural Networks* **5**(1): 130–147.
- Bäck, T., Fogel, D. B. & Michalewicz, T. (2000). *Evolutionary Computation 1*, Taylor & Francis Group, New York, NY, U.S.A.
- Bertling, L., Miranda, V., Matos, M. A. & Hilber, P. (2007). Multiobjective optimization applied to maintenance policy for electrical networks, *IEEE Transactions on Power Systems* .
- Billinton, R. & Billinton, J. (1989). Selective maintenance schedule of distribution networks based on risk management approach, *IEEE Transactions on Power Delivery* **4**(1): 561–568.
- Brown, R. & Willis, H. (2006). The economics of aging infrastructure, *IEEE power & energy magazine* pp. 36–43.
- Castro, L. N. (2006). *Fundamentals of Natural Computing (Chapman & Hall/Crc Computer and Information Sciences)*, Chapman & Hall/CRC.
- Cavellucci, C., França, P. M., Lyra, C. & Reis, P. A. (2007). *Estratégia Hierárquica de Otimização para Alocação de Recursos de Manutenção*, Relatório Técnico R2 - Atualização do estado do conhecimento, Faculdade de Engenharia Elétrica e de Computação - Universidade Estadual de Campinas.
- Cavellucci, C., González, J. F. V. & Lyra, C. (2006). *Estratégia Hierárquica de Otimização para Alocação de Recursos de Manutenção*, Relatório Técnico R1 - Estudo do Processo

e Definição dos Componentes do Sistema, Faculdade de Engenharia Elétrica e de Computação - Universidade Estadual de Campinas.

- Dolan, E. D. & Moré, J. (2002). Benchmarking optimization software with performance profiles, *Mathematical Programming* **91**(2): 201–213.
- Dominelli, N., Rao, A. & Kundur, P. (2006). Life extension and condition assessment, *IEEE power & energy magazine* pp. 25–35.
- Endrenyi, J. & Anders, G. (2006). Aging, maintenance and reliability, *IEEE power & energy magazine* pp. 59–67.
- Endrenyi, J., Anders, G. & Leite da Silva, A. (1998). Probabilistic evaluation of the effect of maintenance on reliability. an application, *IEEE Transactions on Power Systems* **13**(2): 576–583.
- Feo, T. & Resende, M. (1995). Greedy randomized adaptive search procedures, *Journal of Global Optimization* **6**: 109–133.
- Goffe, W., Ferrier, G. & Rogers, J. (1994). Global optimization of statistical functions with simulated annealing, *Journal of Econometrics* **60**: 65–99.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley.
- Hilber, P. (2008). *Maintenance Optimization for Power Distribution Systems*, PhD thesis, Royal Institute of Technology Stockholm, Sweden.
- Hilber, P. & Bertling, L. (2008). Multiobjective maintenance policy for a distribution system an application study, *PMAFS 2008*.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*, The University of Michigan Press.
- IEEE Standard 493 (1998). IEEE recommended practice for the design of reliable industrial and commercial power systems, *IEEE Gold Book*.
- Janjic, A. & Popovic, D. (2007). Selective maintenance schedule of distribution networks based on risk management approach, *IEEE Transactions on Power Systems* **22**(2): 597–604.
- Kirkpatrick, S., Gelatt, C. D. J. & Vecchi, M. P. (1983). Optimization by simulated annealing, *Science* **220**: 671–680.

- Kuntz, P., Christie, R. & Venkata, S. (2002). Optimal vegetation maintenance scheduling of overhead electric power distribution systems, *IEEE Transactions on Power Delivery* **17**(4): 1164–1169.
- Li, W., Vaahedi, E. & Choudhury, P. (2006). Power system equipment aging, *IEEE power & energy magazine* pp. 52–58.
- Li, Z. & Guo, J. (2006). Wisdom about age, *IEEE power & energy magazine* pp. 44–51.
- Luger, G. B. & Stubblefield, W. A. (1993). *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, The Benjamin/Cummings Publishing Company., Redwood City, California, U.S.A.
- Metropolis, N., Rosenbluth, A., Rosenbluth, M. N., Teller, A. & Teller, E. (1953). Equations of state calculations by fast computing machines, *Journal of Chemical Physics* **21**: 1087–1092.
- Michalewicz, Z. (1996). *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, New York.
- Moubray, J. (1997). *Reliability-Centered Maintenance*, second edition edn, Industrial Press Inc.
- Overman, R. & Collard, R. (2003). The complimentary roles of reliability-centered maintenance and condition monitoring, *18th International Maintenance Conference* .
- Pearl, J. (1984). *Intelligent Search Strategies for Computer Problem Solving*, Addison-Wesley Publishing Company., Los Angeles, California, U.S.A.
- Reis, P. A. (2007). *Otimização baseada em confiabilidade de planos de manutenção de sistemas de distribuição de energia elétrica*, Mestrado, Faculdade de Engenharia Elétrica e de Computação - Universidade Estadual de Campinas, Campinas - SP - Brazil.
- Russell, S. & Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*, 2nd edn, Prentice-Hall, Englewood Cliffs, NJ.
- Siqueira, I. P. (2005). *Manutenção Centrada na Confiabilidade: Manual de Implementação*, Qualitymark Editora LTDA., Rio de Janeiro, RJ, Brasil.
- Sittithumwat, A., Soudi, F. & Tomsovic, K. (2004). Optimal allocation of distribution maintenance resources with limited information, *Electric Power Systems Research* **68**: 208 – 220.