

**UNIVERSIDADE FEDERAL DE ALFENAS**  
**INSTITUTO DE CIÊNCIAS EXATAS**  
**BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

***Tiago Silveira***

**A UTILIZAÇÃO DA META-HEURÍSTICA DE OTIMIZAÇÃO  
DENOMINADA COLÔNIA DE FORMIGAS PARA O  
RESTABELECIMENTO DE SISTEMAS ELÉTRICOS DE  
POTÊNCIA**

Alfenas, 30 de Junho de 2010.



**UNIVERSIDADE FEDERAL DE ALFENAS**  
**INSTITUTO DE CIÊNCIAS EXATAS**  
**BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**A UTILIZAÇÃO DA META-HEURÍSTICA DE OTIMIZAÇÃO  
DENOMINADA COLÔNIA DE FORMIGAS PARA O  
RESTABELECIMENTO DE SISTEMAS ELÉTRICOS DE  
POTÊNCIA**

***Tiago Silveira***

Monografia apresentada ao Curso de Bacharelado em  
Ciência da Computação da Universidade Federal de  
Alfenas como requisito parcial para obtenção do Título de  
Bacharel em Ciência da Computação.

Orientador: Prof. Luiz Eduardo da Silva.

Alfenas, 30 de Junho de 2010.



***Tiago Silveira***

**A UTILIZAÇÃO DA META-HEURÍSTICA DE OTIMIZAÇÃO  
DENOMINADA COLÔNIA DE FORMIGAS PARA O  
RESTABELECIMENTO DE SISTEMAS ELÉTRICOS DE  
POTÊNCIA**

A Banca examinadora abaixo-assinada aprova a monografia apresentada como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação pela Universidade Federal de Alfenas.

---

**Prof. Eliseu César Miguel**

**Universidade Federal de Alfenas**

---

**Prof. Humberto César Brandão de Oliveira**

**Universidade Federal de Alfenas**

---

**Prof. Luiz Eduardo da Silva (Orientador)**

**Universidade Federal de Alfenas**

Alfenas, 30 de Junho de 2010.



Dedico este trabalho a toda minha família, pelo enorme amor e carinho prestados e pelos valiosos ensinamentos que continuarão sendo a base sólida de minha vida.





# AGRADECIMENTOS

Primeiramente, agradeço a Deus, presença constante em minha vida, por tudo que me fora proporcionado e por permitir a conclusão do desenvolvimento deste trabalho.

Ao professor Luiz Eduardo da Silva, orientador deste projeto, que tanto me motivou e me ajudou no desenvolvimento deste trabalho, além dos valiosos conselhos e ensinamentos que pude receber em nossos trabalhos durante toda a minha formação.

Ao professor Humberto César Brandão de Oliveira, co-orientador deste projeto, pela paciência e apoio prestado durante todos os anos de minha graduação, além da estimada amizade por nós desenvolvida.

Ao professor Eliseu César Miguel, pelo esforço incansável em conduzir o curso de Bacharelado em Ciência da Computação da UNIFAL-MG a uma qualidade em busca da excelência.

Ao professor Flávio Barbieri, pelo ensinamento didático prestado.

Ao professor Tomás Dias Sant'Ana, pelo modo simples e único de se fazer algo que se gosta de uma forma bem feita.

A todos meus amigos de sala e demais professores que me ajudaram durante toda esta caminhada.

Por fim, agradeço à UNIFAL-MG pelo apoio financeiro, através do programa PROBIC, para o desenvolvimento deste projeto.



"A mente que se abre a uma nova ideia jamais voltará ao seu tamanho original."

Albert Einstein



# RESUMO

A Otimização por Colônia de Formigas (ACO) é uma nova meta-heurística que imita o comportamento de uma população de agentes (formigas) na procura de alimento. Através do uso de mecanismos de cooperação e adaptação, esta técnica emula a natureza como forma de obter soluções promissoras com ideias simples. Com isso, a ACO vem se mostrando como uma abordagem competitiva em relação a outras estratégias apresentadas na literatura. Em relação à área de aplicação desta técnica, possui um domínio vasto, podendo ser aplicada em vários problemas, estando entre os mais conhecidos, por exemplo, o Problema do Caixeiro Viajante, o Problema de Geração de Horários e o Problema de Roteamento de Veículos, que são problemas de computação difícil, ou seja, problemas de complexidade exponencial. Além destes problemas descritos, um problema bastante conhecido na área de Engenharia Elétrica é o Problema de Restabelecimento de Sistemas Elétricos de Potência. Neste problema, busca-se o restabelecimento de um sistema elétrico, que é feito através da definição de um caminho alternativo da fonte de geração até as cargas que tiveram o atendimento de energia interrompido por uma falha. Este caminho alternativo é definido através da mudança de estado de chaves normalmente abertas e chaves normalmente fechadas do sistema. O sistema tem um número considerável de chaves que podem ser utilizadas na definição deste caminho, o que torna o restabelecimento uma tarefa de computação difícil, sendo fortemente indicado o uso de técnicas de otimização qualificadas para este processo. Assim, a ACO, através de sua estratégia *Max-Min Ant System*, será utilizada para testar a ordem de religamento de chaves de um sistema elétrico, de modo que se faça o menor número de mudança de chaves para o restabelecimento de energia, resultando assim em uma maior economia de recursos necessários para essa tarefa. Para verificar a eficácia do método proposto, durante o desenvolvimento deste trabalho foram realizados testes de comparação com os resultados de outras implementações presentes na literatura, confirmando a ACO com uma promissora alternativa quando aplicada em problemas combinatórios da área da Engenharia Elétrica.

**Palavras-Chave:** Otimização Combinatória, Otimização por Colônia de Formigas, Restabelecimento de Sistemas Elétricos de Potência, Computação Evolucionária.



# ABSTRACT

The Ant Colony Optimization (ACO) is a new meta-heuristic that mimics the behavior of a population of agents (ants) in search of food. Through the use of cooperation and adaptation mechanisms, this technique emulates nature as a way to obtain promising solutions with simple ideas. With this, the ACO presents itself as a competitive approach in relation to other strategies presented in the literature. Regarding the area of application of this technique, has a wide domain and can be applied in various problems, being among the best known, for example, the Traveling Salesman Problem, the Generation Scheduling Problem and the Vehicles Routing Problem, which are difficult computing problems, ie, problems of exponential complexity. Besides these problems described, a problem well known in the area of Electrical Engineering is the Restoration Problem in Electrical Power Systems. In this problem, we seek to restore a power system, which is done by defining an alternate path from the source of generation to the loads that had the power service interrupted by a failure. This alternate path is defined by change of state of normally open switches and normally closed switches of the system. The system has a considerable number of switches that can be used in defining this path, which makes the restoration a task of difficult computing, being strongly indicated the use of qualified optimization techniques to this process. Thus, the ACO, through its strategy *Max-Min Ant System*, will be used to test the order of reclosing of switches of an electric system, to make the fewest changes of the switches to restoration of energy, thus resulting in a greater economy of resources needed for this task. To check the effectiveness of the proposed method, during the development of this work were performed tests for comparison with results from other implementations in the literature, confirming the ACO with a promising alternative when applied to combinatorial problems in the field of Electrical Engineering.

**Keywords:** Combinatorial Optimization, Ant Colony Optimization, Restoration of Electrical Power Systems, Evolutionary Computation.





# LISTA DE FIGURAS

|  |    |
|--|----|
| FIGURA 1 – CONFIGURAÇÃO DO EXPERIMENTO DA PONTE BINÁRIA, ONDE A PONTE RAMIFICA-SE EM COMPRIMENTOS IGUAIS. ....   | 34 |
| FIGURA 2 – CAMINHO TOMADO PELAS FORMIGAS APÓS DETERMINADO TEMPO DE EXPLORAÇÃO DO AMBIENTE, ONDE, PELAS FLUTUAÇÕES ALEATÓRIAS, (A) TOMAM O CAMINHO SUPERIOR E (B) TOMAM O CAMINHO INFERIOR..... | 35 |
| FIGURA 3 – CONFIGURAÇÃO DO EXPERIMENTO DA PONTE BINÁRIA, ONDE A PONTE RAMIFICA-SE EM COMPRIMENTOS DIFERENTES. ....   | 36 |
| FIGURA 4 – CAMINHO TOMADO PELAS FORMIGAS APÓS DETERMINADO TEMPO DE EXPLORAÇÃO DO AMBIENTE.....   | 37 |
| FIGURA 5 – REPRESENTAÇÃO DE UM SISTEMA ELÉTRICO DE POTÊNCIA (A), ASSIM COMO SITUAÇÕES DE RESTABELECIMENTO (B,C,D) EM CASO DE FALHAS NO SISTEMA. ....   | 52 |
| FIGURA 6 – FLUXOGRAMA DA METODOLOGIA PARA O DESENVOLVIMENTO DO SISTEMA. ....   | 60 |
| FIGURA 7 – REPRESENTAÇÃO DA ÁRVORE GERADORA PARA O SEP, SENDO (A) O PRÓPRIO SEP E (B) A SUA ÁRVORE GERADORA. ....  | 65 |
| FIGURA 8 – REPRESENTAÇÃO DA ÁRVORE REVERSA PARA O SEP, NO QUAL (A) MOSTRA SUA REPRESENTAÇÃO GRÁFICA E (B) SUA REPRESENTAÇÃO ESTRUTURAL.....  | 66 |
| FIGURA 9 – INSTÂNCIA " <i>DISTRIBUTION_SYSTEM_01</i> ". ....   | 68 |
| FIGURA 10 – INTERAÇÃO ENTRE OS PARÂMETROS A E B. ....  | 96 |
| FIGURA 11 – INTERAÇÃO ENTRE OS PARÂMETROS A E P. ....  | 96 |
| FIGURA 12 – INTERAÇÃO ENTRE OS PARÂMETROS P E B.....   | 97 |



# LISTA DE TABELAS

|   |    |
|---|----|
| TABELA 1 – VALORES FIXOS ESTABELECIDOS PARA A ACO.....  | 70 |
| TABELA 2 – VALORES VARIÁVEIS ESTABELECIDOS PARA A ACO.....  | 70 |
| TABELA 3 – RESULTADO PARA VARIAÇÃO DO PARÂMETRO <i>NÚMERO DE FORMIGAS</i> .....   | 75 |
| TABELA 4 – RESULTADOS DE SIMULAÇÕES DE FALHAS – NÚMERO DE FECHAMENTOS DE CHAVES NO<br>MAIOR QUE 1.....                            | 78 |
| TABELA 5 – RESULTADOS DE SIMULAÇÕES DE FALHAS – NÚMERO DE FECHAMENTO DE CHAVES NO<br>IGUAL A 1, APRESENTANDO VÁRIAS SOLUÇÕES..... | 79 |
| TABELA 6 – RESULTADOS DE SIMULAÇÕES DE FALHAS – NÚMERO DE FECHAMENTO DE CHAVES NO<br>IGUAL A 1, APRESENTANDO ÚNICA SOLUÇÃO.....   | 80 |
| TABELA 7 – RESULTADOS DE OUTRAS ABORDAGENS PARA FALHAS COM VÁRIAS ALTERAÇÕES DE<br>CHAVES.....                                    | 82 |
| TABELA 8 – RESULTADOS DE OUTRAS ABORDAGENS PARA FALHAS COM VÁRIAS SOLUÇÕES, SENDO<br>POSSÍVEL UMA ÚNICA ALTERAÇÃO DE CHAVE.....   | 83 |
| TABELA 9 – RESULTADOS DE OUTRAS ABORDAGENS COM ÚNICA SOLUÇÃO.....   | 84 |
| TABELA 10 – VALORES PARAMÉTRICOS INICIAIS.....  | 95 |
| TABELA 11 – MELHORES VALORES PARAMÉTRICOS.....  | 95 |
| TABELA 12 – TEMPO MÉDIO DE EXECUÇÃO DOS MELHORES VALORES PARAMÉTRICOS.....  | 97 |



# LISTA DE ABREVIações

|             |                                     |
|-------------|-------------------------------------|
| ACO         | Otimização por Colônia de Formigas  |
| AG          | Árvore Geradora                     |
| AS          | Ant System                          |
| GA          | Algoritmo Genético                  |
| <i>MMAS</i> | Max-Min Ant System                  |
| NO          | Normalmente Aberta                  |
| NC          | Normalmente Fechada                 |
| PSO         | Otimização por Enxame de Partículas |
| RAM         | Memória de Acesso Aleatório         |
| SEP         | Sistema Elétrico de Potência        |
| TSP         | Problema do Caixeiro-Viajante       |



# SUMÁRIO

|   |           |
|---|-----------|
| <b>1 INTRODUÇÃO .....</b>   | <b>25</b> |
| 1.1 JUSTIFICATIVA E MOTIVAÇÃO.....                                | 26        |
| 1.2 PROBLEMATIZAÇÃO .....   | 27        |
| 1.3 OBJETIVOS .....   | 27        |
| 1.3.1 Gerais .....  | 27        |
| 1.3.2 Específicos.....  | 28        |
| 1.4 ORGANIZAÇÃO DA MONOGRAFIA.....                                | 28        |
| <b>2 A META-HEURÍSTICA ANT COLONY OPTIMIZATION.....</b>           | <b>31</b> |
| 2.1 CONSIDERAÇÕES INICIAIS .....                                  | 31        |
| 2.2 A META-HEURÍSTICA .....                                       | 32        |
| 2.3 A INSPIRAÇÃO BIOLÓGICA .....                                  | 32        |
| 2.3.1 O Experimento da Ponte Binária .....                        | 34        |
| 2.4 A TÉCNICA DE OTIMIZAÇÃO .....                                 | 37        |
| 2.4.1 Representação do Problema.....                              | 37        |
| 2.4.2 Formigas Artificiais .....                                  | 39        |
| 2.4.3 Atualização do Feromônio.....                               | 41        |
| 2.5 IMPLEMENTAÇÃO DA META-HEURÍSTICA ACO .....                    | 43        |
| 2.6 IDEALIZAÇÃO DA META-HEURÍSTICA ACO .....                      | 44        |
| 2.7 VARIAÇÕES DA META-HEURÍSTICA ACO.....                         | 46        |
| 2.7.1 <i>Max-Min Ant System (MMAS)</i> .....                      | 46        |
| 2.7.2 Outras Estratégias Bem-Sucedidas.....                       | 47        |
| 2.8 CONSIDERAÇÕES FINAIS.....                                     | 48        |
| <b>3 RESTABELECIMENTO DE SISTEMAS ELÉTRICOS DE POTÊNCIA .....</b> | <b>49</b> |
| 3.1 CONSIDERAÇÕES INICIAIS .....                                  | 49        |
| 3.2 SISTEMAS ELÉTRICOS DE POTÊNCIA .....                          | 50        |
| 3.3 RESTABELECIMENTO DE SISTEMAS ELÉTRICOS DE POTÊNCIA .....      | 51        |
| 3.4 CONSIDERAÇÕES FINAIS.....                                     | 53        |
| <b>4 REVISÃO BIBLIOGRÁFICA.....</b>                               | <b>55</b> |
| 4.1 CONSIDERAÇÕES INICIAIS .....                                  | 55        |
| 4.2 TRABALHOS COM BASES DISTINTAS.....                            | 56        |
| 4.3 TRABALHO COM BASE SIMILAR .....                               | 57        |
| 4.4 CONSIDERAÇÕES FINAIS.....                                     | 58        |
| <b>5 PROPOSTA.....</b>  | <b>59</b> |
| 5.1 CONSIDERAÇÕES INICIAIS .....                                  | 59        |
| 5.2 METODOLOGIA.....  | 59        |
| 5.3 ARQUITETURA DO SISTEMA .....                                  | 61        |
| 5.3.1 Formigas.....   | 61        |
| 5.3.2 Informação Heurística.....                                  | 62        |
| 5.3.3 Feromônio .....   | 63        |
| 5.4 ESTRUTURAS DE DADOS ADICIONAIS .....                          | 64        |
| 5.4.1 Árvore Geradora.....  | 64        |

|          |  |           |
|----------|--|-----------|
| 5.4.2    | Árvore Reversa.....  | 66        |
| 5.5      | CONFIGURAÇÕES DO SISTEMA.....                                | 67        |
| 5.5.1    | SEP.....   | 68        |
| 5.5.2    | MMAS.....  | 69        |
| 5.5.3    | Demais Configurações.....                                    | 71        |
| 5.6      | CONSIDERAÇÕES FINAIS.....                                    | 71        |
| <b>6</b> | <b>RESULTADOS.....</b>                                       | <b>73</b> |
| 6.1      | CONSIDERAÇÕES INICIAIS.....                                  | 73        |
| 6.2      | EXPERIMENTO PARA O PARÂMETRO <i>NÚMERO DE FORMIGAS</i> ..... | 74        |
| 6.3      | EXPERIMENTO COM OS PARÂMETROS ESTABELECIDOS.....             | 77        |
| 6.4      | COMPARAÇÃO COM OUTRAS ABORDAGENS.....                        | 81        |
| 6.5      | CONSIDERAÇÕES FINAIS.....                                    | 85        |
| <b>7</b> | <b>CONCLUSÕES.....</b>                                       | <b>87</b> |
| 7.1      | CONCLUSÕES GERAIS.....                                       | 87        |
| 7.2      | TRABALHOS FUTUROS.....                                       | 88        |
| <b>8</b> | <b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>                       | <b>89</b> |
| <b>9</b> | <b>APÊNDICE I.....</b>                                       | <b>93</b> |
| 9.1      | CONSIDERAÇÕES INICIAIS.....                                  | 93        |
| 9.2      | PARÂMETROS.....  | 93        |
| 9.3      | AJUSTES PARAMÉTRICOS.....                                    | 94        |
| 9.4      | CONSIDERAÇÕES FINAIS.....                                    | 97        |



# 1

## Introdução

*Este capítulo apresenta os detalhes do desenvolvimento do trabalho, mostrando seus objetivos e metas, sua justificativa e a sua forma de organização.*

Atualmente, pesquisadores de todo o mundo propõem novos métodos para resolver problemas clássicos/complexos de forma mais simples e/ou eficientes. Uma simples prova disso são as novas técnicas de otimização baseadas em inteligência de enxames, onde através da cooperação entre os indivíduos (de forma direta ou indireta) consegue-se uma melhor adaptação da população ao ambiente.

Um exemplo de uma técnica de inteligência de enxames é a Otimização por Colônia de Formigas (ACO), inspirada no comportamento de agentes (formigas artificiais) em busca de alimento. A Otimização por Colônia de Formigas, originalmente descrita por Dorigo (1992), tem como ideia primordial a comunicação indireta entre seus indivíduos, através de uma trilha feita por cada formiga durante a exploração do espaço de busca. Esta trilha é feita utilizando-se uma espécie de feromônio artificial, que atua como um atrativo para as mesmas, servindo como uma informação perceptível que é modificada pelas formigas para refletir sua experiência de busca atual. Assim, a ACO mostra-se como um processo de aprendizado distribuído em que simples agentes, as formigas, trabalham de forma coletiva, modificando, adaptativamente, a forma como o problema é representado e percebido por outras formigas (DORIGO; STÜTZLE, 2002). Toda a robustez e eficiência da ACO giram em torno dessa ideia.

## 1.1 Justificativa e Motivação

Hoje em dia, a demanda por energia elétrica vem crescendo em todos os setores que necessitam desta para suas atividades. Com isso, torna-se fundamental a elaboração de sistemas que façam a distribuição da energia produzida que sejam confiáveis e disponíveis. Estes são os chamados Sistemas Elétricos de Potência (SEPs). Deste modo, torna-se indispensável também a elaboração de um serviço de qualidade que manipule tais sistemas, de forma a fornecer esta energia sempre de forma confiável, disponibilizado-a em níveis de máxima margem de segurança possível (MANSOUR, 2009).

Diante desta característica, a utilização de métodos computacionais para a manipulação destes sistemas, a fim de garantir uma maior disponibilidade e confiabilidade, é cada vez mais comum, devido aos enormes benefícios proporcionados, como a rapidez, segurança e economia para sua operação. Entre os métodos para a manipulação destes sistemas, pode-se utilizar, por exemplo, a Busca Tabu (GLOVER, F., 1989; GLOVER, F. 1990), o *Simulated Annealing* (KIRKPATRICK; GELATT; VECCHI, 1983) e os algoritmos evolucionários, com as técnicas baseadas em inteligência de enxames, como a Otimização por Enxame de Partículas (KENNEDY; EBERHART, 1995) e a meta-heurística de Otimização por Colônia de Formigas ou ACO (do inglês, *Ant Colony Optimization*) (DORIGO, 1992), que será a base para o desenvolvimento deste trabalho.

Por sua proposta original de trabalhar com problemas de caráter combinatório, a ACO é indicada para proceder com a manipulação de sistemas elétricos de potência, pois esta tarefa está relacionada a um problema desta natureza, sendo conhecido como Problema de Restabelecimento de Sistemas Elétricos de Potência.

Em um problema de Restabelecimento de Sistemas Elétricos de Potência, o restabelecimento é feito através da definição de um caminho alternativo da fonte de geração até as cargas que tiveram o atendimento de energia interrompido por uma falha. Este caminho alternativo é definido através de mudança de estado de chaves normalmente abertas e chaves normalmente fechadas do sistema. O sistema

tem um número considerável de chaves que podem ser utilizadas na definição deste caminho, o que torna o restabelecimento uma tarefa de complexidade exponencial. Logo, a adoção de uma heurística de otimização, como a ACO, é indispensável para se obter uma resolução factível em um tempo viável.

Por fim, a naturalidade em resolver um problema, tendo como base as ideias da natureza, é um fator interessante que faz parte da meta-heurística ACO. Dessa forma, este foi um dos motivos para a escolha desta pesquisa como Trabalho de Conclusão de Curso, pois tal técnica de otimização apresenta uma ideia bastante simples, mas que ao mesmo tempo gera um atrativo e motivação para a busca de novas pesquisas nesta área.

## **1.2 Problematização**

Em relação ao problema de restabelecimento de sistemas elétricos de potência surgem os seguintes questionamentos: como otimizar os gastos decorrentes de um falha ou pane de um determinado sistema elétrico ao procedermos com a mudança de estados das chaves desse sistema para o restabelecimento de energia para os pontos afetados? Existe uma ordem pré-determinada de mudança de estados das chaves para o restabelecimento de energia de um sistema elétrico que diminua os gastos necessários para esse processo? Como tratar o problema de sobrecarga nas linhas do sistema?

## **1.3 Objetivos**

### **1.3.1 Gerais**

Aplicar ao problema de restabelecimento de sistemas elétricos de potência uma recente meta-heurística de otimização combinatória, a chamada Otimização por Colônia de Formigas, sendo esta uma técnica que segue princípios relativamente

simples, mas ao mesmo tempo mostra-se bastante eficiente se aplicada a problemas dessa natureza. O intuito desse trabalho é reduzir os custos para a resolução do problema elencado, através da obtenção da menor sequência de alterações de chaves pelo algoritmo da meta-heurística ACO.

### **1.3.2 Específicos**

- Implementar o algoritmo para a resolução do problema de restabelecimento de sistemas elétricos de potência em linguagem de programação C#;
- Realizar testes com o modelo idealizado para a coleta de dados;
- Realizar testes estatísticos com os dados obtidos do algoritmo;
- Avaliar o desempenho da meta-heurística proposta para a resolução do problema descrito;
- Comparar os resultados obtidos com outros resultados presentes na literatura;
- Elaborar artigos científicos para a submissão em eventos correlatos.

## **1.4 Organização da Monografia**

Este trabalho está estruturado da seguinte maneira: o Capítulo 2 apresenta os principais conceitos relacionados à meta-heurística ACO, que será fundamental para o entendimento da forma que o problema do SEP será otimizado; o Capítulo 3 apresenta a descrição básica do problema de SEPs; o Capítulo 4 apresenta alguns trabalhos que trataram o problema do SEP com a utilização de métodos computacionais bio-inspirados para a sua resolução; o Capítulo 5 apresenta a forma de mapeamento de um SEP para o modelo de feromônio da ACO, as estruturas de dados adicionais utilizadas e as configurações adotadas, tanto para a instância do

problema, quanto para a arquitetura da ACO, além das demais configurações utilizadas; o Capítulo 6 apresenta os resultados obtidos para as configurações experimentais que foram adotadas para a resolução do problema, bem como as comparações com outros resultados obtidos na literatura; e, finalmente, o Capítulo 7 apresenta as conclusões obtidas no desenvolvimento do presente trabalho, buscando demonstrar seus reais benefícios para problemas reais de SEPs; neste, os possíveis trabalhos futuros também são apresentados.



## 2

# A Meta-Heurística *Ant Colony Optimization*

*Este capítulo apresenta os principais conceitos relacionados à técnica de otimização denominada Ant Colony Optimization, utilizada no trabalho. O capítulo retrata os aspectos da concepção à atualidade desta metodologia, desde a inspiração biológica, até a sua evolução para algoritmos mais robustos que hoje são largamente utilizados em problemas combinatórios.*

### 2.1 Considerações Iniciais

Hoje em dia, muitos problemas de otimização numérica apresentam uma grande dificuldade para serem resolvidos, tanto no domínio de soluções contínuas quanto no de soluções discretas. Isso se deve ao fato da quantidade de soluções possíveis que um problema pode ter em seu espaço de busca, sendo, geralmente, uma quantidade exponencial em relação à instância do problema (problemas NP<sup>1</sup> - completo). Assim, muitas vezes, a busca pela melhor solução de um determinado problema não será viável para o tempo disponível e um resultado aproximado talvez seja a melhor escolha naquele momento.

Diante desta característica, o desenvolvimento de técnicas de otimização numérica que trabalhem na busca de qualificadas soluções em tempo factível se faz necessário. Apesar de inúmeras técnicas bem-sucedidas existentes, este trabalho irá focar exclusivamente em uma meta-heurística para otimização combinatória denominada Otimização por Colônia de Formigas ou ACO (do inglês, *Ant Colony Optimization*) (DORIGO; STÜTZLE, 2002).

xxxí \_\_\_\_\_

<sup>1</sup> *Non-deterministic Polynomial* - algoritmos exponenciais no tempo de execução têm sua função de complexidade descrita da forma  $O(c^n)$ , onde  $c > 1$ . Este grupo contém problemas cujos melhores algoritmos determinísticos conhecidos são não-polinomiais (CORMEN *et al.*, 2002).

## 2.2 A Meta-Heurística

No início da década de 1990, a ACO foi introduzida por Marco Dorigo (DORIGO, 1992), em sua tese de doutorado, como uma nova meta-heurística para a resolução de difíceis problemas de otimização combinatória.

A ACO é uma técnica de propósito geral de inteligência de enxames. Os algoritmos de inteligência de enxames são feitos de simples indivíduos que cooperam através da auto-organização, ou seja, sem qualquer forma de controle central sobre os membros do enxame (DORIGO; SOCHA, 2007). Devido ao seu promissor desempenho, torna-se assim um dos exemplos mais bem-sucedidos de sistemas de inteligência de enxames (BONABEAU; DORIGO; THÉRAULAZ, 1999) e, com isso, tem sido aplicada em vários tipos de problemas, desde os problemas de natureza estática (no qual a topologia e o custo do problema não mudam durante a otimização), como o clássico Problema do Caixeiro Viajante, até os problemas de natureza dinâmica (no qual a topologia e o custo podem mudar ao longo da otimização), como o Problema de Roteamento em Redes de Telecomunicações.

Fornecendo uma definição matemática, a ACO é englobada como uma meta-heurística baseada em probabilidade, onde, por sua proposta original de trabalhar com um conjunto de soluções discretas, envolve basicamente uma busca de caminho em grafos, sendo, dessa forma, possível construir suas soluções ao longo do processo de otimização.

Em seguida, serão apresentados os principais conceitos relacionados a origem e implementação da meta-heurística ACO.

## 2.3 A Inspiração Biológica

Em meados do século 20, o entomologista francês Pierre-Paul Grassé (GRASSÉ, 1946 *apud* DORIGO; BIRATTARI; STÜTZLE, 2006) observou que algumas espécies de cupins, durante a vida na colônia, produziam uma substância que afetava o comportamento dos indivíduos desta organização. Grassé verificou que a



substância produzida tinha uma relação direta com a forma de comunicação entre esses insetos, pois tinham uma reação diferente ao entrar em contato com esta substância. Tal reação dos cupins passou a ser estudada por Grassé, que foi por ele denominada de "*significant stimuli*" (estímulo significante).

Em seus estudos, Grassé verificou que o efeito causado pela reação podia atuar como um novo estímulo para os insetos que produziam a referida substância e para os outros insetos da colônia. Com isso, utilizou o termo 'estigmergia' para descrever esse tipo de comunicação feita por esses insetos (GRASSÉ, 1959 *apud* DORIGO; BIRATTARI; STÜTZLE, 2006).

A estigmergia possui algumas características que a diferencia de outros tipos de comunicação. Segundo Dorigo, Birattari e Stütze (2006), as principais características são:

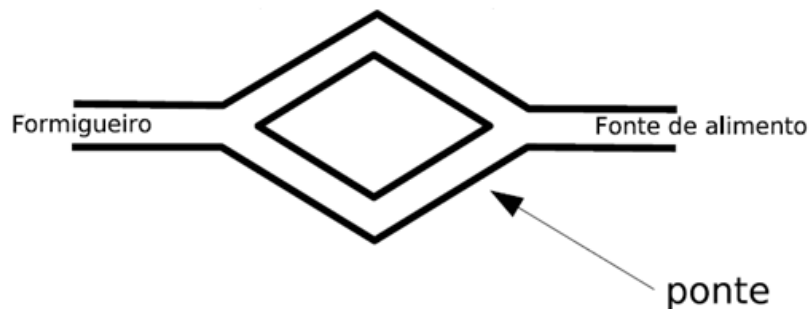
- Forma de comunicação indireta e não-simbólica mediada pelo ambiente, onde os insetos trocam informações, modificando o seu ambiente;
- Informação estritamente local, onde pode somente ser acessada pelos insetos que visitam o local onde ela foi depositada.

Esse tipo de comunicação pode ser observado em outros indivíduos. Como exemplo, se tem a comunicação feita por formigas em uma colônia. Muitas espécies de formigas caminham na direção do seu formigueiro a uma fonte de alimento (e vice-versa) deixando no caminho uma substância atrativa chamada feromônio. Com isso, as formigas conseguem perceber a presença do feromônio deixado por outras formigas, fazendo com que estas tendam a seguir aqueles caminhos com uma concentração maior de feromônio, por motivos biológicos. Logo, esse mecanismo é a forma efetiva que as formigas têm para procurar e transportar o alimento encontrado para o seu formigueiro (DORIGO; BIRATTARI; STÜTZLE, 2006).

### 2.3.1 O Experimento da Ponte Binária

Diante do comportamento das formigas por sua forma de comunicação, Deneubourg *et al.* (1990) *apud* Dorigo, Birattari, Stützle (2006) estudaram seus procedimentos de depósito de feromônio. Para isso, foram realizados experimentos com uma colônia de formigas, sendo estas, formigas argentinas reais (*Iridomyrmex humilis*), onde tal experimento ficou conhecido como Experimento da Ponte Binária.

O Experimento da Ponte Binária tem a seguinte configuração: existem dois extremos, no qual um é a colônia de formigas e o outro uma fonte de alimento. Para que as formigas tenham acesso a esse alimento, há uma conexão através de uma ponte entre a colônia e a fonte de alimento. Porém, tal ponte tem uma configuração especial, pois esta sofre uma bifurcação e, em seguida, uma junção destas pontes, apresentando dois caminhos de igual comprimento, similar à Figura 1.



**Figura 1 – Configuração do experimento da ponte binária, onde a ponte ramifica-se em comprimentos iguais.**

Colocadas nessa configuração, as formigas começavam a explorar os arredores do formigueiro, atravessavam a ponte e eventualmente chegavam à fonte de alimento. Note que para uma formiga que sai do formigueiro consiga chegar à fonte de alimento, ela obrigatoriamente tem que passar por uma das ramificações da ponte.

Durante todo esse processo de exploração, as formigas vão depositando o feromônio para manter a orientação do caminho tomado em seu percurso. Inicialmente, como não há nenhum vestígio de feromônio em qualquer parte do percurso, cada formiga escolhe aleatoriamente uma das duas pontes. Entretanto, devido oscilações aleatórias de probabilidade, visto que as duas pontes têm o mesmo tamanho, depois de algum tempo uma destas irá ter um acúmulo maior de feromônio e, com isso, irá atrair mais formigas para o seu percurso. Assim, o resultado será que, ao final, toda a colônia irá convergir para aquele percurso que tornou-se mais atrativo, fazendo com que o outro caminho não seja utilizado. Devido a igualdade de probabilidade de escolha dos caminhos, a colônia irá escolher cerca de 50% cada uma das pontes durante a busca pelo alimento (DENEUBOURG *et al.*, 1990 *apud* DORIGO; BIRATTARI; STÜTZLE, 2006). A Figura 2 representa a escolha tomada pela colônia durante esse ciclo de exploração.



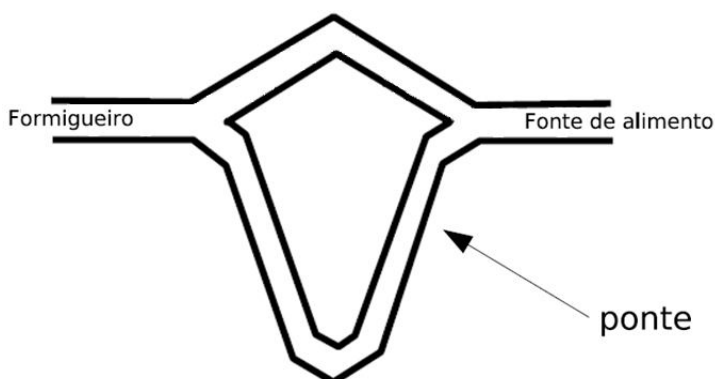
**Figura 2 – Caminho tomado pelas formigas após determinado tempo de exploração do ambiente, onde, pelas flutuações aleatórias, (a) tomam o caminho superior e (b) tomam o caminho inferior.**

Assim, esse comportamento da colônia permitiu a Deneubourg *et al.* ultimarem que as formigas assumiam um comportamento interessante. Enquanto uma formiga isolada realiza movimentações essencialmente aleatórias, uma formiga que encontra um caminho previamente percorrido pode detectá-lo e então decidir com uma maior probabilidade em segui-lo, reforçando, assim, o caminho com o seu próprio feromônio. Essa decisão feita por uma formiga segue um comportamento autocatalítico de retroalimentação positiva, ou seja, quanto mais formigas adotam um caminho, mais atrativo este se torna, aumentando também a sua probabilidade de escolha.

Com isso, pode ser adotada uma estratégia útil com essa característica das formigas: o uso de um comportamento autocatalítico pode ser explorado para fazer

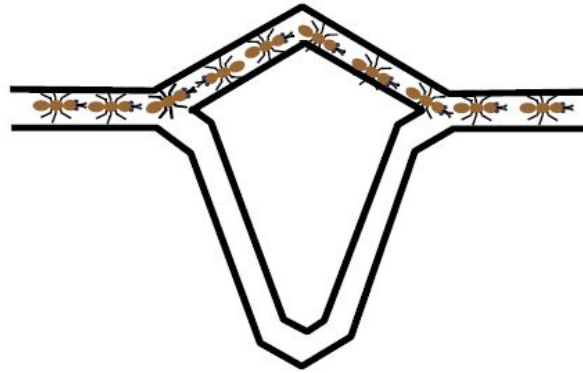
com que as formigas encontrem o caminho mais curto entre o seu formigueiro e a fonte de alimento. Esta hipótese foi comprovada pelo experimento realizado por Goss *et al.* (1989) *apud* Dorigo, Birattari, Stützle (2006), ao aplicar uma variação no experimento da ponte binária original de Deneubourg.

Em seu experimento, Goss *et al.* seguiram basicamente a mesma ideia do experimento de Deneubourg *et al.*, porém considerando que uma das pontes que se bifurcam entre o formigueiro e a fonte de alimento é significativamente maior que a outra. A configuração do experimento de Goss *et al.* pode ser observada na Figura 3.



**Figura 3 – Configuração do experimento da ponte binária, onde a ponte ramifica-se em comprimentos diferentes.**

Agora, as oscilações estocásticas na escolha inicial de uma das pontes por uma formiga são muito reduzidas devido um fator importante: aquelas formigas que tiveram a sorte de escolher o caminho mais curto serão as primeiras a retornar ao formigueiro, por motivos lógicos. Assim, quanto mais uma formiga realiza esse percurso, mais rápido será o seu retorno e maior será a quantidade de feromônio nesse caminho em relação ao caminho longo, fazendo com que a probabilidade de mais formigas escolherem o caminho curto seja maior (DORIGO; BIRATTARI; STÜTZLE, 2006). Após algum tempo, a probabilidade de escolher o caminho mais curto será tão alta, devido a quantidade de feromônio neste caminho, que raramente uma formiga irá eleger a outra ponte para realizar o seu percurso. Ao final, a ponte com o caminho mais curto será o caminho escolhido por todas as formigas, obtendo a configuração similar a Figura 4.



**Figura 4 – Caminho tomado pelas formigas após determinado tempo de exploração do ambiente.**

Assim sendo, com base em todas as características apresentadas do comportamento real de algumas formigas na busca de alimento, Dorigo (1992) propôs um algoritmo que assume exatamente este comportamento, onde sua utilização tornou-se bastante atraente e utilizada para a resolução de muitos problemas computacionais combinatórios de complexidade exponencial.

## **2.4 A Técnica de Otimização**

O modelo idealizado por Goss *et al.* (1989) *apud* Dorigo, Birattari, Stützle (2006) que explica o comportamento forrageiro das formigas foi a principal fonte utilizada por Dorigo para o desenvolvimento da otimização por colônia de formigas (DORIGO; BIRATTARI; STÜTZLE, 2006). Modelando o problema real para um modelo matemático, algumas definições foram formuladas, que serão apresentadas e discutidas a seguir.

### **2.4.1 Representação do Problema**

A principal característica da meta-heurística ACO em seu funcionamento é em relação ao seu tipo de estrutura utilizada para a otimização. Para isso, baseando-se no comportamento real das formigas, a ponte binária foi substituída por uma

representação em grafos e as trilhas de feromônio por trilhas de feromônio artificiais.

Adotando uma abordagem formal, segundo Dorigo e Stützle (2002), têm-se as seguintes definições: considerando um problema de minimização  $(\xi, f, \Omega)$ , onde  $\xi$  é o conjunto de soluções candidatas,  $f$  é a função objetivo que avalia uma possível solução candidata  $s \in \xi$ , atribuindo-lhe um custo, e  $\Omega$  o conjunto de restrições definidas para o problema que as soluções devem cumprir. A meta para o problema é encontrar a solução ótima global  $s_{best} \in \xi$ , ou seja, aquela solução de menor custo que satisfaça todas as restrições  $\Omega$ .

Definindo uma representação para um problema de otimização combinatória, de acordo com a abordagem anterior, são apresentadas as seguintes características:

- Um conjunto finito  $C = \{c_1, c_2, \dots, c_n\}$  de componentes, que representam os nós ou vértices do grafo;
- O estado do problema é definido pelas sequências  $x = \{c_i, c_j, \dots, c_k, \dots\}$ , onde o conjunto de todas as sequências possíveis é denotado por  $X$ ;
- O conjunto finito de restrições  $\Omega$  define o conjunto de estados possíveis  $\mathcal{G}$ , onde  $\mathcal{G} \subseteq X$ ;
- Um conjunto  $\xi^*$  de soluções possíveis, com  $\xi^* \subseteq \mathcal{G}$  e  $\xi^* \subseteq \xi$ ;
- Um custo  $f(s, t)$  é associado a cada solução candidata  $s \in \xi$ .

Considerando esta representação, as formigas da meta-heurística ACO buscam soluções através da construção de caminhos no grafo  $G = (C, A)$ , onde os vértices ( $C$ ) são os componentes do problema e as arestas ( $A$ ) são as ligações que existem entre estes componentes. A implementação das restrições pode ser feita na forma de restrição absoluta, excluindo a ligação entre os componentes do problema que determinam sequências de soluções inconsistentes, ou então as restrições

podem ser implementadas na forma de restrição de preferência, na qual a ligação é possível, mas implica na penalização da solução.

Assim, a forma de modelar um problema de otimização combinatória utilizando uma abordagem da ACO será associá-lo a um modelo de feromônio, de forma que seja emulado o comportamento de formigas reais. Com isso é possível fazer que um valor de feromônio seja associado a cada solução componente durante a otimização (DORIGO; BIRATTARI; STÜTZLE, 2006), dando ao comportamento da execução do algoritmo, a similaridade com o modelo real.

## 2.4.2 Formigas Artificiais

As formigas artificiais são heurísticas construtivas representadas por estruturas de dados associadas ao problema e por procedimentos que determinam como uma solução deve ser construída através de um percurso no grafo  $G = (C, A)$  do problema, que desenvolvem a construção de uma solução iterativamente, através das decisões probabilísticas tomadas durante esse processo.

As formigas constroem soluções consistentes ou inviáveis, através de uma função derivada das restrições do problema. Uma solução completa é formada adicionando possíveis soluções componentes a soluções parciais Para isso, leva em consideração duas informações elementares (DORIGO; STÜTZLE, 2002):

- Trilha de feromônio artificial: assim como em um comportamento real de uma formiga, as formigas artificiais também utilizam uma ideia similar para indicar possíveis caminhos interessantes para a construção de uma solução, com a utilização da trilha de feromônio artificial. Esta trilha irá mudar dinamicamente durante a execução do algoritmo para refletir a experiência de busca adquirida pelas formigas durante a construção das soluções;
- Informação heurística: outro ponto fundamental utilizado na meta-heurística ACO, que corresponde a um valor específico do problema a ser resolvido, se este valor estiver disponível.

Com essas informações, as decisões tomadas pelas formigas não serão escolhas arbitrárias durante a otimização, mas terão uma política baseada em probabilidade, onde aquelas soluções componentes com maior influência dessas informações terão uma maior probabilidade de serem adicionadas a cada iteração à solução parcial que está sendo construída. Assim, utilizando essas informações para tomar uma decisão probabilística, as formigas são capazes de construir uma grande variedade de qualificadas soluções, podendo, assim, explorar uma vasta quantidade destas e, com isso, aumentar as chances de encontrar soluções mais promissoras para o problema em que são aplicadas.

Mais precisamente, cada formiga  $k$  da colônia tem as seguintes propriedades:

- Ela explora o grafo  $G = (C, A)$  para buscar por soluções viáveis  $s$  de custo mínimo. Ou seja, soluções  $s$  tal que  $\hat{f}_s = \min_s f(s, t)$ ;
- Ela tem uma memória  $M^k$  usada para guardar a informação sobre o caminho parcial. A memória pode ser usada (i) para construir soluções viáveis, (ii) para avaliar a solução encontrada, e (iii) para refazer o caminho inverso para depositar feromônio;
- Ela pode ser atribuída um estado inicial  $x_s^k$ , definido com uma sequência de comprimento unitário (componente simples ou sequência vazia) e uma ou mais condições de parada  $e^k$ ;
- Quando estiver em um estado  $x_r = \langle x_{r-1}, i \rangle$ , ela tenta mover para qualquer nó  $j$  em sua vizinhança viável  $N_i^k$ , que é um estado  $\langle x_r, j \rangle \in \mathcal{G}$ . Se isso não é possível, então a formiga pode mover para um nó  $j$  nessa vizinhança inviável  $IN_i^k$ , gerando, dessa forma, um estado inviável;
- Ela seleciona o movimento através da aplicação de uma regra de decisão probabilística, que considera as (i) trilhas de feromônio



disponíveis localmente e valores heurísticos, (ii) a memória privada da formiga de nós visitados e (iii) as restrições do problema.

- O procedimento de construção da formiga  $k$  para quando pelo menos uma das condições de parada  $e^k$  é satisfeita.
- Ao adicionar um componente  $c_j$  à solução atual, ela pode atualizar a trilha de feromônio associada a esta conexão correspondente.
- Uma vez construída uma solução, ela pode refazer o mesmo caminho inverso e atualizar as trilhas de feromônio das conexões usadas.

Uma importante característica observada por Dorigo e Stützle (2002) sobre as formigas artificiais segue o mesmo princípio do comportamento de uma formiga real. Durante a construção de uma solução, ao longo do processo de otimização, as formigas movem-se simultaneamente e independentemente, sendo capazes até (apesar de uma baixa probabilidade) de encontrar uma boa solução para o problema em questão. Porém, será através da interação com outras formigas artificiais durante a busca, que este processo de construção estará mais propenso a encontrar soluções satisfatórias.

### 2.4.3 Atualização do Feromônio

Outro aspecto importante acerca do funcionamento da meta-heurística da ACO é a atualização que o feromônio artificial sofre durante o processo de otimização. Dado o grafo de busca  $G = (C, A)$  do problema, durante ou após a construção de uma solução por uma formiga, ocorrem dois eventos a respeito das trilhas de feromônio:

- Depósito: processo em que a trilha de feromônio de uma solução componente é incrementada com um valor referente a qualidade que o caminho ou aresta encontrada terá sobre as próximas iterações;
- Evaporação: processo no qual a intensidade das trilhas de feromônio vai diminuindo ao longo do tempo.

Ambos os eventos são cruciais na ACO, tanto que os principais algoritmos que se baseiam nesta meta-heurística se diferenciam, principalmente, por variações na utilização e atualização destes termos, conforme será visto posteriormente.

O depósito de feromônio é importante para qualificar aqueles caminhos que vem se destacando como os mais promissores, fazendo com que tenham uma maior influência na decisão das formigas nas iterações posteriores, como já foi discutido.

A evaporação do feromônio é o outro aspecto importante sobre o feromônio utilizado pelas formigas. Sua importância mais evidente é que ela não permite que a quantidade de feromônio cresça indefinidamente, o que poderia causar a estagnação das formigas em um mínimo local, pois um caminho iria acumular tanto feromônio e seria tão influente, que outros caminhos dificilmente teriam a possibilidade de serem escolhidos. Todavia, esse processo emula um mecanismo de “esquecimento” de soluções anteriores, permitindo a exploração de novas áreas do espaço de busca (DORIGO; STÜTZLE, 2002). Ao mesmo tempo em que permite o “esquecimento” de boas soluções, a evaporação do feromônio também permite o “esquecimento” de más soluções, fazendo com que uma decisão ruim tomada no início não influencie negativamente a busca por melhores soluções durante todo o processo de otimização.

Basicamente, apesar das variações algorítmicas baseadas na meta-heurística ACO existentes, a quantidade de feromônio depositado pelas formigas artificiais durante a otimização será da seguinte forma: seja  $\Delta\tau_{ij}^k(t)$  definido como a quantidade de feromônio que a formiga  $k$  deposita sobre a trilha entre  $i$  e  $j$  na iteração  $t$ . Assim, a atualização será dada pela Equação 1.

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k(t)}, & \text{se a } k\text{-ésima formiga usa a trilha } (i, j) \text{ no seu percurso;} \\ 0, & \text{caso contrário.} \end{cases} \quad (1)$$

em que  $Q$  é uma constante ( $Q > 0$ ), e  $L_k(t)$  é o tamanho do percurso da  $k$ -ésima formiga na iteração  $t$ . Logo, como a ideia original das trilhas de feromônio, a Equação 1 mostra que quanto menor a distância do percurso  $(i, j)$  da formiga, mais

feromônio vai ser recebido no arco  $(i,j)$  pertencente ao percurso, aumentando a sua probabilidade em escolhas futuras.

## 2.5 Implementação da Meta-Heurística ACO

A meta-heurística ACO pode ser resumida da seguinte forma: as formigas da colônia, de forma assíncrona e concorrente, controem soluções para o problema modelado através da descrição de um caminho no grafo  $G = (C, A)$  que representa o problema. A escolha de cada formiga é feita através de uma decisão probabilística, levando-se em consideração a trilha de feromônios e a informação heurística. Durante o processo de construção ou depois que a formiga finalizou um percurso no grafo  $G$ , as formigas da colônia podem avaliar a solução que foi construída e, assim, depositar feromônio no caminho a fim de privilegiar a melhor solução encontrada pela colônia.

De uma forma simplificada, a meta-heurística ACO pode ser descrita através do Algoritmo 1:

```
meta_heurística_ACO {
    inicializações()
    enquanto (!condicaoDeTermino()) {
        constroiSoluções()
        atualizaEstatísticas()
        executaPesquisaLocal()
    }
}
```

### Algoritmo 1 – Pseudo-código de meta-heurística ACO.

A rotina denominada `inicializações()`, compõe as operações que devem ser realizadas no início do processo de construção. Nesta rotina, estão implementadas operações como criação das estruturas para representar as instâncias do problema e a colônia de formigas. A repetição delimita as três operações fundamentais realizadas neste algoritmo. Não está definido explicitamente neste pseudo-código, mas estas operações podem ser realizadas em

paralelo. Esta repetição acontece até que a rotina `condiçãoDeTermino()` obtenha sucesso. A condição de término pode ser o tempo máximo determinado para busca ou então até que um parâmetro de qualidade da solução tenha sido alcançado. As operações fundamentais da meta-heurística ACO são: a construção de soluções pelas formigas da colônia, a atualização de estatísticas e estruturas utilizadas na implementação da meta-heurística e a execução de alguma operação adicional, como a execução do algoritmo de Busca Local, a fim de melhorar a solução encontrada. Esta última operação é opcional, mas pode determinar a qualidade e a velocidade de convergência da meta-heurística para uma solução. Na rotina `constroiSoluções()`, cada formiga da colônia, a partir de um componente aleatório do problema, constrói uma solução completa. Em função das soluções encontradas, podem ser executadas algumas operações adicionais a fim de premiar as boas soluções encontradas e penalizar as soluções ruins. Na rotina `atualizaEstatística()` são realizadas estas operações.

Seguindo esta estrutura elementar, vários algoritmos foram propostos na literatura, sendo alguns apresentados a seguir.

## 2.6 Idealização da Meta-Heurística ACO

Baseado na estrutura original do funcionamento da meta-heurística ACO, foi proposto por Dorigo um algoritmo que simula a colônia de formigas na busca de alimentos para resolver o problema de otimização combinatória, denominado Problema do Caixeiro Viajante (JÜNGER; REINELT; RINALDI, 1997). A primeira aplicação feita com um algoritmo de otimização por colônia de formigas foi feita usando este problema como teste (DORIGO; CARO; GAMBARDELLA, 1998). Tal problema teve grande importância para o desenvolvimento da meta-heurística ACO, por ser um bom *benchmark* para testar e comparar as variações construídas desta técnica.

O algoritmo original para a meta-heurística ACO foi denominado AS (*Ant System*). Para utilizar o AS, o problema a ser resolvido deve ser modelado através de um grafo, conforme já discutido, onde os vértices representam estados do

problema e as arestas determinam o custo da ligação entre os vértices. A tarefa das formigas é encontrar um caminho mais curto entre um vértice inicial e um vértice final do grafo. Cada formiga, ao se mover no grafo, espalha feromônio no seu caminho. A formiga possui a memória do caminho já percorrido e dos vértices já visitados, com o intuito de evitar passar pelo mesmo vértice mais de uma vez. Quando a formiga está no vértice  $i$ , ela escolhe o próximo vértice  $j$  usando a função de probabilidade, conforme Equação 2:

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta} \quad (2)$$

no qual  $\tau_{ij}(t)$  determina a quantidade de feromônio entre os vértices  $i$  e  $j$  na iteração  $t$ ; a variável  $\eta_{ij}$  representa a função heurística que é específica para cada problema;  $\alpha$  e  $\beta$  determinam a relevância da trilha de feromônio e da informação heurística, respectivamente; e  $l$  representa a vizinhança ainda não visitada do vértice  $i$ , para toda a vizinhança factível  $N$  da formiga  $k$ .

Além disso, a trilha de feromônios evapora por uma constante  $\rho$ , conforme Equação 3, para evitar a convergência de todas as formigas para um caminho sub-ótimo.

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k(t), \quad \forall(i, j) \quad (3)$$

onde  $m$  é número de formigas e  $\Delta\tau_{ij}(t)$  representa a quantidade de feromônio que a formiga  $k$  deixa entre os vértices  $i$  e  $j$  na iteração  $t$ .

Este processo de escolha de caminhos para a construção de soluções e atualização de feromônio é repetido até que todas as formigas tenham encontrado um caminho. Em seguida, procede-se com atualização das trilhas de feromônio, levando em consideração a relevância do caminho construído, sendo este processo feito por todas as formigas. O algoritmo termina depois de  $n$  iterações, ou até que um tempo limite tenha sido alcançado.

Ao final deste processo, o melhor caminho encontrado entre todas as iterações é escolhido como a melhor solução encontrada pela colônia de formigas.

## 2.7 Variações da Meta-Heurística ACO

Conforme dito anteriormente, o AS foi o primeiro algoritmo desenvolvido que seguia as ideias da meta-heurística da ACO. Depois dele, diversas variações deste algoritmo original foram desenvolvidas, com o intuito de melhorar o desempenho e obter melhores resultados.

Em seguida, algumas das estratégias da ACO são apresentadas.

### 2.7.1 *Max-Min Ant System (MMAS)*

Uma das implementações mais bem sucedidas da meta-heurística ACO é o algoritmo denominado *Max-Min Ant System* (STÜTZLE; HOOS, 2000). As duas principais inovações nesta implementação são que somente a melhor formiga atualiza a trilha de feromônio e que o nível de feromônio de cada trilha está limitado pelas variáveis  $\tau_{\max}$  e  $\tau_{\min}$ , isto é, o feromônio destas não pode ser superior a  $\tau_{\max}$  nem inferior a  $\tau_{\min}$ . A atualização do feromônio nesta implementação é dada pela Equação 4.

$$\tau_{ij}(t+1) = [(1 - \rho) \cdot \tau_{ij}(t) + \Delta \tau_{ij}^{best}]_{\tau_{\min}}^{\tau_{\max}} \quad (4)$$

em que  $\Delta \tau_{ij}^{best}$  representa as arestas do caminho construído pela melhor formiga da colônia, e os colchetes na equação delimitam o valor de resposta para atualização do feromônio. Os valores de  $\tau_{\max}$  e  $\tau_{\min}$  podem ser definidos empiricamente ou através de valores específicos do problema. A melhor formiga pode ser a melhor formiga da iteração ou a melhor formiga geral.

Assim, adotando apenas estas modificações na abordagem original da ACO, esta se tornou uma de suas estratégias mais importantes. Contudo, outras estratégias também possuem suas características e marcam significativamente a

evolução desta meta-heurística, pois sempre foram propostas como alternativas para a busca por um melhor desempenho na resolução de um problema.

## 2.7.2 Outras Estratégias Bem-Sucedidas

Além da estratégia *MMAS*, apresentada anteriormente, outras estratégias se destacaram durante o desenvolvimento da meta-heurística ACO. Entre as variações, temos os algoritmos como o *Ant System* baseado numa estratégia elitista (DORIGO, 1992; DORIGO; MANIEZZO; COLORNI, 1996), o *Ant System* baseado em *rank* ( $AS_{rank}$ ) (BULLNHEIMER; HARTL; STRAUSS, 1999), *Hyper-Cube Ant System* (HC-ACO) (BLUM; ROLI; DORIGO, 2001) e *Ant Colony System* (ACS) (DORIGO; GAMBARDELLA, 1997).

A variação baseada em estratégia elitista consiste em premiar a melhor solução construída com uma quantidade adicional de feromônio. O  $AS_{rank}$  é, de certa forma, uma variação do algoritmo elitista, onde somente um *rank* das melhores soluções é permitido fazer a atualização das trilhas de feromônio, com um taxa proporcional a qualidade da solução. O *Hyper-Cube Ant System* aparece como uma variação do *MMAS*, definindo valores máximos e mínimos para a trilha de feromônios no intervalo  $[0,1]$ , utilizando uma regra de atualização de feromônios baseada na média ponderada de vetores de feromônio dentro do espaço de um cubo com arestas de tamanho 1, reiniciando as trilhas a toda estagnação em mínimos locais. E o *Ant Colony System*, que introduziu a possibilidade de atualização de feromônio local, além da atualização realizada ao final do processo de construção.

Como visto, cada uma das variações dos algoritmos da ACO tem a sua característica determinante. Isso permite que, dependendo do tipo de problema a se aplicar uma determinada estratégia, se possa ter uma melhor adaptação desta e, conseqüentemente, apresentar um melhor desempenho sobre tal problema, quando utilizada uma estratégia específica.

## 2.8 Considerações Finais

O campo de pesquisa na área de otimização combinatória e, especialmente, na meta-heurística ACO é vasto, permitindo que outras contribuições ainda possam ser feitas no sentido de produzir boas extensões do algoritmo original.

A utilização da meta-heurística ACO nos diversos tipos de problema de otimização combinatória é outro campo importante de pesquisa. No entanto, novas pesquisas ainda são necessárias a fim de consolidar esta ferramenta de inteligência artificial como uma importante ferramenta para resolução de problemas combinatórios de computação difícil, no qual um dos papéis deste presente trabalho busca contribuir.



# 3

# Restabelecimento de Sistemas Elétricos de Potência

*Este capítulo aborda os principais aspectos relacionados a SEPs, tais como seus principais componentes, características e possíveis problemas que podem acontecer durante o seu funcionamento.*

## 3.1 Considerações Iniciais

Atualmente, a demanda por energia elétrica vem crescendo em todos os setores que necessitam desta para suas atividades. Com isso, torna-se indispensável elaborar um serviço de qualidade que forneça esta energia de forma mais confiável possível. Isto significa que a energia produzida deve ser entregue aos consumidores de forma contínua, de modo que se tenha uma tensão constante em níveis de máxima margem de segurança possível (MANSOUR, 2009).

Diante desta característica, é inquestionável a elaboração de sistemas, os quais façam esta distribuição, que sejam também confiáveis e disponíveis no maior grau possível. Estes são os chamados Sistemas Elétricos de Potência. Assim, a utilização de métodos computacionais para manipulação destes sistemas que garantam estas características são cada vez mais comuns, devido aos enormes benefícios proporcionados, como a rapidez, segurança e economia para sua operação.

Visando proporcionar um melhor entendimento de como o método computacional que será elaborado neste trabalho será utilizado para a manipulação de um SEP, primeiramente serão abordadas algumas características dos SEPs, que auxiliarão no entendimento do problema.

## 3.2 Sistemas Elétricos de Potência

Sistemas Elétricos de Potência (SEPs) consistem em sistemas que englobam as unidades de geração, transmissão e a distribuição da energia produzida (ZANETTA JR., 2005). Entretanto, apesar da grande importância de cada uma dessas unidades, este trabalho irá focar apenas nos assuntos referentes à unidade de transmissão de um SEP. Basicamente, a unidade de transmissão aborda todos os meios físicos existentes (linhas de transmissão) de modo que a energia produzida na unidade geradora seja entregue à unidade de distribuição e possíveis consumidores desta energia.

Atualmente, com o aumento da demanda por energia elétrica, torna-se fundamental também a demanda por sistemas elétricos de potência que sejam confiáveis e disponíveis (LIU; GU, 2006; ROSSI, 2000). Porém, é impossível garantir que um SEP estará sempre disponível, devido a possibilidade de ocorrência de falhas em partes do sistema em algum momento.

Uma falha em um SEP corresponde à quebra da configuração atual desse sistema, podendo fazer com que algumas cargas (ou seja, unidades que demandam energia) tenham a sua alimentação interrompida. Logo, falhas causadas por danos físicos requerem que alguns componentes destes sistemas sejam reparados ou substituídos, para que estes retornem a sua atividade normal, ou seja, a um estado de distribuição de energia às cargas similar ao anterior à falha.

Contudo, enquanto os componentes afetados por falhas não são substituídos, a arquitetura de um SEP, para que esta tenha uma maior confiabilidade, possui ligações alternativas entre as cargas, permitindo diversas configurações operacionais, isto é, variações no percurso original são possíveis para a alimentação destas cargas. Espera-se, então, que o caminho das linhas de transmissão de um SEP sejam reconfiguradas, de modo que se retome o atendimento às cargas o mais rápido possível, ao menor custo.

Com isso, algumas características devem ser seguidas e respeitadas em um SEP durante a atividade de correção de uma falha para o restabelecimento de

energia destes sistemas. Tais características estão embutidas no problema de restabelecimento de SEPs, e serão discutidas.

### **3.3 Restabelecimento de Sistemas Elétricos de Potência**

Diante da ocorrência de uma falha, a fim de garantir uma redução mínima na confiabilidade do sistema, as áreas isoladas por tal falha deverão ser realimentadas com energia. A tarefa de reconfiguração deste sistema para atingir este objetivo é dado o nome de Serviço de Restabelecimento (WATANABE, 2005).

O serviço de restabelecimento envolve escolher uma sequência de chaves (elementos capazes de fazer a interligação entre linhas de transmissão) a fim de estabelecer um caminho alternativo entre a fonte geradora e as cargas que tiveram o serviço de atendimento interrompido, para o restabelecimento do sistema, após a ocorrência de uma falha. Para isso, o SEP conta com linhas alternativas entre as cargas que possuem chaves normalmente abertas (desconectam uma linha de energia do sistema), sendo as primeiras a terem o seu estado alterado no caso de uma falha. Em situações normais, as cargas possuem chaves que estão normalmente fechadas (conectam uma linha de energia do sistema), sendo estas as primeiras que têm o seu estado alterado em caso de uma sobrecarga de alguma linha do sistema.

Para o serviço restabelecimento de um SEP, devem ser observadas as seguintes características (WATANABE, 2005):

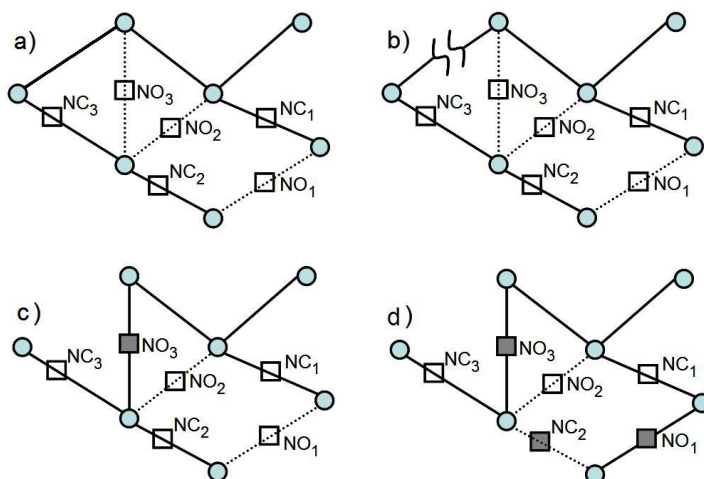
- Restrição radial da rede: devido topologia de um SEP ser similar a de uma Árvore Geradora, o número de linhas que transferem energia, de modo que todas as cargas sejam atendidas, deve sempre ser igual ao número de cargas menos um;
- Restrição da fonte de energia: as cargas totais do sistema não devem ultrapassar o limite máximo da fonte de energia correspondente.

Geralmente, esta restrição é mais importante quando mais de uma unidade geradora alimenta um SEP;

- Restrição da capacidade da linha: o fluxo de energia por uma linha não deve ser maior que sua capacidade. Tratada em todos os casos, sendo esta a restrição que eleva o grau do problema de restabelecimento de SEPs de um problema polinomial para um problema de complexidade exponencial (problema NP-Completo).

Tais características são obrigatórias em um SEP que apresenta uma topologia radial. Uma característica desejável é que todas as cargas que tiveram o atendimento interrompido por uma falha sejam supridas após o serviço de restabelecimento. Porém, isso pode depender da configuração da rede.

Todo o processo descrito acima é mostrado na Figura 5. A Figura 5 (a) ilustra um SEP em seu funcionamento normal, com algumas chaves normalmente abertas ( $NO_i$ ), para o restabelecimento do sistema em caso de falhas, e as outras fechadas ( $NC_i$ ), para que todas as cargas sejam supridas. A Figura 5 (b) ilustra uma situação de falha. A Figura 5 (c) ilustra uma possível solução para o restabelecimento do sistema, onde fechando apenas a chave  $NO_3$ , todas as cargas têm o seu atendimento garantido. Por fim, a Figura 5 (d) ilustra uma possível solução, caso a linha com a chave  $NC_2$  da Figura 5 (c) tenha uma sobrecarga.



**Figura 5 – Representação de um Sistema Elétrico de Potência (a), assim como situações de restabelecimento (b,c,d) em caso de falhas no sistema.**

Um fato que pode ser observado pela Figura 5 é sobre o número de chaves alteradas no caso de uma falha. Devido a topologia da rede, para que o seu aspecto radial seja mantido, o número de chaves que tem o seu estado alterado é sempre ímpar. Outro ponto fundamental para o restabelecimento é que para realizar esse processo, a sequência de chaves deve ser intercalada entre chave normalmente aberta e chave normalmente fechada.

Assim, à medida que um SEP vai aumentando a sua complexidade, o número de chaves desse sistema cresce consideravelmente, a fim de manter a sua disponibilidade e confiabilidade. Porém, com um número maior de chaves, o número de sequências possíveis na tentativa de um restabelecimento torna-se muito grande, devido ao número de combinações possíveis com tais chaves para formar uma sequência que satisfaça as restrições do problema. Logo, todo esse processo poderá ser dispendioso, o que torna o restabelecimento de sistemas elétricos de potência uma tarefa de computação difícil, devido à sua complexidade exponencial de possibilidades para proceder com um restabelecimento.

### **3.4 Considerações Finais**

Os SEPs constituem hoje fontes de grande importância para qualquer atividade desenvolvida na sociedade. Logo, fazer com que estes se tornem mais seguros e confiáveis é a meta que se tem buscado.

Assim, uma das formas de alcançar estas características é adicionar várias linhas alternativas ao SEP radial, que poderão ser utilizadas no caso de uma falha que venha a ocorrer, evitando que parte deste sistema fique sem energia, desde que se tenha um serviço de restabelecimento eficaz. Contudo, pelas várias restrições impostas nestes sistemas, ele acaba se tornando um problema multi-restritivo, onde não é importante somente encontrar uma sequência de chaves que torne a restabelecer a energia momentaneamente, mas uma sequência que satisfaça todas as cargas, não causando sobrecarga em outras linhas e respeitando a capacidade máxima da fonte de geração.

Assim, para que seja cumprido o objetivo deste trabalho, a aplicação de uma técnica de otimização, a ACO, foi proposta para proceder com o restabelecimento de SEPs em caso de falhas. Esta será o cerne para a resolução do problema apresentado.

# 4

## Revisão Bibliográfica

*Este capítulo apresenta uma revisão bibliográfica sobre o tema abordado neste trabalho. Trata-se de uma verificação das abordagens computacionais que já foram adotadas para a resolução do problema de restabelecimento de SEPs, com enfoque nos modelos bio-inspirados.*

### 4.1 Considerações Iniciais

A busca por novas formas de resolução de um problema é algo inerente e objetivo primordial de qualquer pesquisador. A partir do momento que se identifica um problema, têm-se duas possibilidades: tal problema não consta de um método para sua resolução, no qual será proposta, então, uma forma de resolução; ou este já contém uma possível técnica aplicada para sua resolução, no qual qualquer pessoa que siga tal área de pesquisa será motivada a pesquisar e desenvolver novas formas para trabalhar com o problema, a fim de obter uma melhor otimização deste.

Diante disso, a área de pesquisa de sistemas elétricos de potência já conta com várias técnicas de resolução, no caso em que ocorram problemas de falhas nos circuitos e seja necessário proceder com o restabelecimento das linhas destes sistemas. Como prova disto, temos os trabalhos de Mansour (2009), Watanabe (2005) e Lambert-Torres *et al.* (2009).

Visando fundamentar o desenvolvimento deste trabalho, destacam-se neste capítulo algumas das principais técnicas bio-inspiradas de otimização para resolução do problema de restabelecimento de SEPs encontradas na literatura. Estas técnicas foram separadas em duas seções, sendo que a primeira mostra as técnicas que utilizaram uma base de testes para seus algoritmos propostos diferente da base utilizada neste trabalho; e a segunda Seção apresenta as técnicas de resolução que levam em consideração a mesma base de teste deste trabalho, sendo possível uma forma de comparação entre estas estratégias.

## 4.2 Trabalhos com Bases Distintas

Os trabalhos apresentados nesta seção são aqueles que utilizam uma base de testes diferente da apresentada neste trabalho, para as respectivas técnicas propostas para restabelecimentos de seus SEPs.

Em Mansour (2009), um Algoritmo Evolucionário para a resolução de problemas de grande porte de sistemas de distribuição radial é proposto. Mais especificamente, é proposta uma melhoria na técnica de otimização destes problemas, ao utilizar uma técnica baseada em Algoritmo Genético Elitista (*Elitist GA*), sendo denominada NS2R. Para um auxílio na otimização, o algoritmo NS2R foi desenvolvido com o uso de RNP (Representação Nó-Profundidade). O NS2R modela o problema de restabelecimento considerando 2 funções objetivo, que são: a minimização de operações de troca de chaves; e a satisfabilidade das restrições do problema. Os resultados apresentados para a técnica desenvolvida mostraram-se bastante eficientes, isto porque a variação dos GAs, na qual foi baseada a implementação, é, segundo os autores, uma das técnicas de busca multi-objetivo que tem apresentado melhor desempenho, quando aplicada aos problemas de restabelecimento de SEPs com a característica radial. Além disso, os resultados são ainda mais otimizados, pelo fato da utilização da Representação Nó-Profundidade, pois possibilita a geração somente de configurações factíveis durante a otimização. Isto é fundamental para o bom desempenho do algoritmo que fora proposto, acelerando o processo de busca de soluções.

Outro trabalho relevante é o de Watanabe (2005). Em seu trabalho, Watanabe propõe um algoritmo baseado na ACO para a minimização da energia não fornecida às cargas do sistema, durante o processo de restabelecimento de um SEP. Como estratégia de otimização da ACO, é utilizada a chamada ACO Hiper-Cubo (HC-ACO), por apresentar bons resultados para problemas geração de horário, no qual o problema tratado poderia ser transformado. Para a obtenção dos resultados, foi apresentada uma interessante análise, levando em consideração a variação do número de formigas e do parâmetro de evaporação das trilhas de feromônio ( $\rho$ ) para a resolução do problema. Mesmo utilizando uma instância pequena para a aplicação de técnicas de otimização, constando de 30 cargas



consumidoras e 40 linhas de transmissão, foram testadas todas as situações de falhas possíveis no sistema, sendo que, para a maioria dos casos, foram obtidas as soluções ótimas em um pequeno intervalo de tempo.

### 4.3 Trabalho com Base Similar

Esta seção apresenta os trabalhos que utilizaram, para a verificação da eficiência de seus algoritmos, a mesma instância do SEP utilizada neste trabalho, a chamada "*Distribution\_System\_01*", que será apresentada no capítulo posterior.

No trabalho de Lambert-Torres *et al.* (2009), uma análise comparativa entre 2 algoritmos da computação evolucionária é feita para o problema de restabelecimento de SEP. Tais algoritmos referem-se aos Algoritmos Genéticos (GA) e a Otimização por Enxame de Partículas (PSO). Ambas as técnicas seguem uma abordagem multi-objetiva, apresentando uma evolução de acordo com o número de cargas supridas, o número de linhas com sobrecarga e o número de ciclos criados. É importante observar que a PSO, em concepção original, foi desenvolvida para a resolução de problemas no âmbito dos números reais, ou seja, em um domínio diferente do problema de restabelecimento de SEP, que possui uma abordagem combinatória. Com isso, o algoritmo da PSO teve de sofrer uma adaptação para que pudesse ser aplicado neste tipo de problema. Em se tratando dos resultados apresentados, observou-se um comportamento distinto entre ambas as técnicas. Apesar da adaptação da PSO para o problema do restabelecimento, esta apresentou resultados mais satisfatórios e significativos em falhas que necessitavam de mais alterações de chaves, ou seja, situações onde mais de uma chave NO precisava ser fechada devido a problemas de sobrecarga. Já para os problemas que necessitavam apenas do fechamento de uma chave NO para sua resolução, o GA mostrou-se mais eficiente em todos os casos, porém com um intervalo de tempo menos relevante para aplicações reais (diferença em centésimos de segundo).

## 4.4 Considerações Finais

Como visto, inúmeras variações entre técnicas bio-inspiradas foram desenvolvidas para tentar melhorar a otimização de problemas de restabelecimento. Cada um segue um processo distinto, modelando o problema de acordo com a forma de representação deste na estrutura do algoritmo adotado.

Assim, diante dos trabalhos apresentados, o algoritmo apresentado neste trabalho será baseado em outra abordagem, que buscará um melhor desempenho para problemas de restabelecimento. Como já dito, será adotada a abordagem da ACO como técnica de otimização, utilizando estruturas de dados adicionais para agilizar a busca de soluções.

# 5

## Proposta

*Este capítulo apresenta a proposta desta monografia. É organizado da seguinte forma: a Seção 5.2 apresenta a metodologia utilizada para a idealização e execução do algoritmo. A Seção 5.3 apresenta a arquitetura geral do sistema, sendo esta forma que o SEP foi modelado para o modelo de feromônio da ACO. A Seção 5.4 apresenta as estruturas adicionais utilizadas, a fim de melhorar o desempenho do algoritmo. Por fim, a Seção 5.5 apresenta as configurações básicas adotadas pela estratégia MMAS, assim como a configuração do SEP e demais configurações utilizadas para a execução dos testes com o modelo idealizado.*

### 5.1 Considerações Iniciais

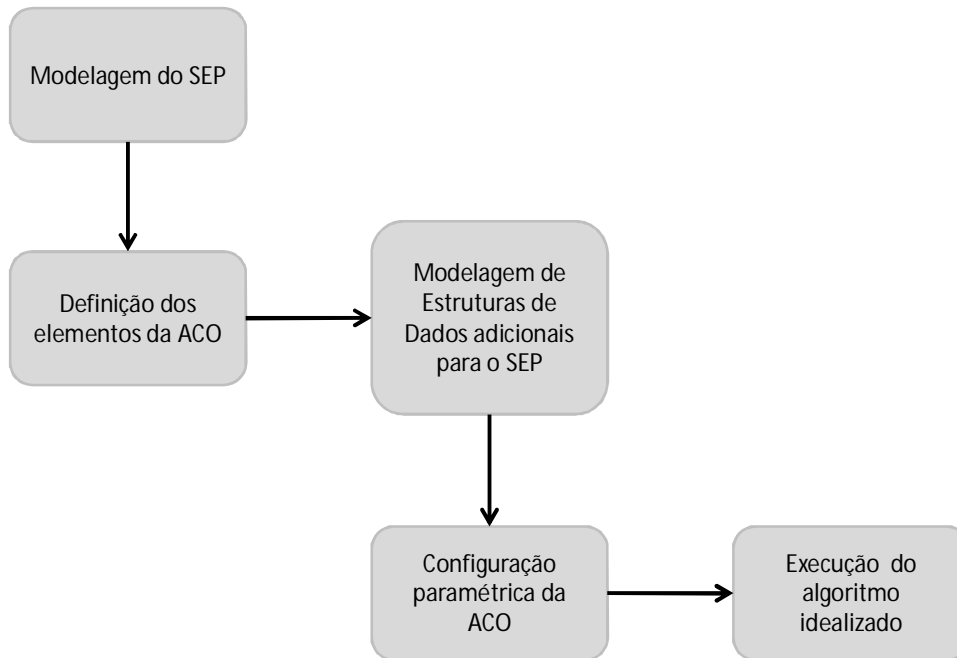
Mediante toda a teoria anteriormente apresentada, será tratada neste capítulo a forma que o sistema foi modelado para trabalhar sobre o problema do restabelecimento de SEPs, através do uso da meta-heurística da ACO.

Assim, o SEP foi idealizado para o modelo de feromônio da ACO, utilizando como estratégia desta a abordagem MMAS, sendo possível, então, alcançar os objetivos propostos.

Alguns aspectos adicionais relacionados à instância do problema foram adotados, a fim de racionalizar o desempenho do algoritmo idealizado, e serão comentados posteriormente.

### 5.2 Metodologia

Para facilitar o entendimento do sistema desenvolvido neste trabalho, são apresentados os passos da metodologia seguida no seu desenvolvimento. A Figura 6 apresenta o fluxograma dos passos adotados para a idealização do sistema.



**Figura 6 – Fluxograma da metodologia para o desenvolvimento do sistema.**

Inicialmente, o SEP foi modelado para uma representação em grafos, para que a ACO pudesse ser aplicada. Em seguida, os elementos essenciais do algoritmo da ACO para a construção de soluções para o problema de restabelecimento de SEPs foram definidos e modelados; esta é a parte elementar do sistema, pois converte o SEP para o modelo de feromônio da ACO. O próximo passo foi modelar as estruturas adicionais para melhorar o desempenho do algoritmo da ACO sobre um problema de restabelecimento. Com todas as etapas anteriores realizadas, a ACO passou por um processo de ajuste paramétrico, a fim de encontrar as melhores configurações para a execução do algoritmo. Ao final, assumindo as configurações paramétricas estabelecidas na etapa anterior, o algoritmo idealizado foi executado sobre uma instância de um SEP, adotando determinadas configurações de *hardware* e *software*, no qual os resultados obtidos foram comparados com outros resultados da literatura.

Em seguida, uma definição mais detalhada para a idealização do modelo é apresentada.

## 5.3 Arquitetura do Sistema

A fim de se mapear o problema do SEP para o modelo de feromônio da ACO, existem algumas características do algoritmo que são essenciais para o seu funcionamento. Entre eles estão as formigas, a informação heurística e o feromônio.

### 5.3.1 Formigas

As formigas da colônia serão a essência da técnica proposta, pois são elas as responsáveis pela construção de uma solução do problema de SEPs em caso de ocorrência de uma falha.

Dessa forma, a construção de uma solução pelas formigas, seguindo a modelagem de feromônio da ACO para a evolução do enxame, foi feita da seguinte forma: na elaboração de uma solução, cada formiga vai possuir uma sequência de chaves  $S = \{C_i, C_{i+1}, \dots, C_n\}$ . Entre estas chaves, estão embutidas os dois tipos de chaves já comentadas, as normalmente abertas (NO) e as normalmente fechadas (NC). Um ponto fundamental para o restabelecimento de um SEP é a característica radial que a rede deve manter, conforme já foi discutido. Com isso, para que esta característica seja sempre satisfeita, é obrigatório a seguinte ordem para o restabelecimento: fechamento de uma chave NO e verificação da condição de construção de solução válida; caso haja sobrecarga, abre-se uma chave NC, fecha-se uma NO e verifica-se novamente a condição de construção de solução válida; esse processo segue até que determinada formiga encontre uma possível solução para o problema. Diante disso, o conjunto  $S$  é ordenado da seguinte forma: para todo  $i$  ímpar do conjunto, o elemento referenciado corresponderá a uma chave NO que poderá ser fechada para a construção de uma solução; e para todo  $i$  par do conjunto, o elemento referenciado corresponderá a uma chave NC, que poderá ser aberta para satisfazer alguma restrição. Seguindo este raciocínio, teremos que a cardinalidade do conjunto  $S$  será sempre ímpar, pois é obrigatória a intercalação de uma sequência par (abertura e fechamento de chaves) ao primeiro fechamento, caso uma única alteração não seja suficiente para encontrar uma resposta para o problema.

Diante disso, a função objetivo para o problema de SEPs torna-se fácil de ser deduzida. Quanto menor for a cardinalidade do conjunto  $S$ , melhor será a solução encontrada. Em outras palavras, deseja-se fazer apenas uma única alteração de chaves no caso de uma falha, correspondendo a solução ótima, com o conjunto  $S$  possuindo cardinalidade igual a 1. Se isso não for possível, busca-se encontrar o menor conjunto possível para se ter uma solução. Assim, a função objetivo é descrita pela Equação 5:

$$f(x) = Tamanho(S)_k \quad (5)$$

no qual  $S$  é o conjunto de chaves alteradas, e  $k$  a uma determinada formiga da colônia. Contudo, apesar da apresentação desta função objetivo, qualquer solução deve antes respeitar todas as restrições para o problema de restabelecimento de SEPs, para que, efetivamente, esta possa ser aplicada.

### 5.3.2 Informação Heurística

A informação heurística é outro ponto importante na construção de uma solução para o problema de SEPs. Como as formigas detêm um processo de construção estocástico e devido a baixa influência das trilhas iniciais de feromônio, guiá-las inicialmente para uma possível solução factível é algo indispensável neste momento.

Um ponto importante que se deve ter consciência para este problema de restabelecimento é que não há qualquer informação mensurável entre as ligações de cargas que sirva para uma escolha de um percurso pelas formigas quando as trilhas de feromônio têm pouca ou nenhuma influência, o que difere, por exemplo, do TSP, que fornece essa informação. Isso acaba gerando uma dificuldade para se estabelecer uma função heurística que seja interessante para este problema.

Assim, a informação heurística cabível para o problema do SEP está relacionada às chaves que devem ser escolhidas para o seu restabelecimento: quais chaves NO que devem ser fechadas e quais chaves NC que devem ser abertas.

Para as chaves NO que devem ser fechadas, a melhor heurística é proceder com o fechamento daquelas mais próximas à região de falha. Isso, pelo motivo de

satisfabilidade: quanto mais próximo à falha uma chave NO for fechada, maior será a chance de se ter um sistema com 100% das cargas restabelecidas. Se por um acaso não houver mais chaves NO próximas à região de falha, certamente este sistema não será mais restabelecido em sua totalidade.

Para as chaves NC que devem ser abertas, este princípio não se sustenta, pois estas não influenciam diretamente na satisfabilidade do sistema, sendo responsáveis apenas por evitar sobrecargas nas linhas do sistema. Logo, a escolha destas chaves será determinada pela experiência da colônia durante a otimização.

Desse modo, estas heurísticas são a base para um bom funcionamento do algoritmo durante a sua execução.

### 5.3.3 Feromônio

Os níveis de feromônio no domínio do problema são uma das partes principais para o funcionamento de um algoritmo que segue a abordagem da ACO. Como já mostrado, o feromônio é um dos responsáveis pela escolha dos caminhos na função de probabilidade que cada formiga vai seguir durante a sua construção de soluções. Porém, novamente, o SEP não possui caminhos entre as cargas nos quais seja possível fazer o depósito de feromônio, pois não se busca a otimização de caminhos, mas a otimização de trocas de chaves.

Logo, os níveis de feromônio estarão relacionados a todas as chaves que este SEP possui, tanto as NO quanto as NC. Formalmente, teremos  $\tau_{NC}[i]$  relacionado à quantidade de feromônio presente na chave fechada NC de índice  $i$ ; e  $\tau_{NO}[j]$  relacionado à quantidade de feromônio depositado na chave aberta NO de índice  $j$ .

Apesar de os níveis de feromônio depositados pela formigas não estarem relacionados aos percursos ou ligações entre as cargas, a ideia de funcionamento deste termo é idêntica ao modelo idealizado da ACO. Assim, quanto mais feromônio possui uma chave do SEP, seja ela qual for (NO ou NC), tem-se a indicação de que esta chave é bastante propícia a ser utilizada no restabelecimento deste, pois várias formigas estão utilizando-a e, com isso, alterando seus níveis de feromônio. Desse modo, quanto maior a quantidade de feromônio de uma chave,

maior a sua probabilidade de escolha na função de probabilidade do algoritmo que as formigas utilizam para construir suas soluções para o problema.

## 5.4 Estruturas de Dados Adicionais

A intenção de utilizar um algoritmo de otimização em um problema real tem a finalidade de se obter uma resposta rápida e interessante para o problema em questão, como pôde ser visto em seções anteriores. Além da pura utilização de um algoritmo que atue na resolução de um problema, pode-se recorrer a algumas estruturas de dados que melhorem a representatividade do problema a ser tratado para a técnica de otimização que será aplicada, de modo a tornar mais natural e específica a representação deste para a estrutura que o algoritmo necessita para sua execução, melhorando, dessa forma, o seu desempenho. Assim, construir uma base na qual as estruturas auxiliem a manipulação do problema, fazendo com que o algoritmo tenha um melhor desempenho, é algo, ao mesmo tempo, interessante e necessário para se buscar resultados mais promissores.

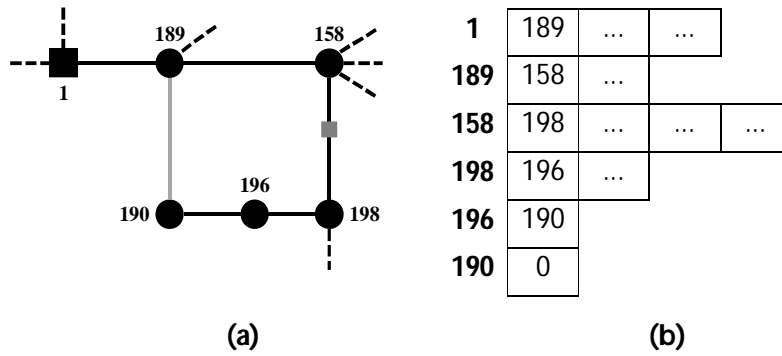
Assim sendo, este trabalho buscou utilizar-se de estruturas de dados adicionais para representação do problema de SEPs para o modelo de feromônio da ACO, fazendo com que a representatividade de tal problema melhorasse a manipulação da construção de soluções pelas formigas e, conseqüentemente, o algoritmo implementado aumentasse sua otimização. Para isso, foram utilizadas duas estruturas de dados referentes a estruturas de grafos, que são: Árvore Geradora e Árvore Reversa.

### 5.4.1 Árvore Geradora

Baseando-se na estrutura geral de uma Árvore Geradora descrita em Cormen *et al.*, (2002), a Árvore Geradora para um SEP pode ser entendida da seguinte maneira: seja um grafo não orientado  $G$  de um SEP, composto pelo conjunto  $C$  de vértices, representando as chaves do sistema, e  $A$  de arestas, representando os caminhos possíveis entre as chaves do conjunto  $C$ . Assim, busca-se encontrar um subconjunto acíclico  $T$  contido ou igual a  $A$  que conecte todos os vértices do sistema. Como o



conjunto  $T$  é acíclico e conecta todos os vértices de  $C$ , temos que tal conjunto forma uma árvore com  $C-1$  arestas, que é chamada de *Árvore Geradora (AG)*. A forma que a AG foi concebida para o sistema pode ser observada na Figura 7.



**Figura 7 – Representação da Árvore Geradora para o SEP, sendo (a) o próprio SEP e (b) a sua Árvore Geradora.**

Pela Figura 7, pode-se observar as seguintes características: a Figura 7 (a) representa uma instância de um SEP qualquer; a Figura 7 (b) modela a AG para o SEP apresentado. Na estrutura da AG da Figura 7 (b), tem-se, para cada linha desta, as cargas do sistema, sendo a primeira a fonte de geração (representada pela carga de número 1); em seguida, para a mesma linha desta estrutura, são representadas as chaves filhas, que são as chaves na qual existe conexão entre a chave inicial e as demais chaves da respectiva linha desta estrutura. A carga que possui ligação para uma de número zero (0), indica que esta é uma carga terminal do sistema. Assim, partindo da fonte de geração, ou seja, da raiz da árvore, é possível chegar às cargas terminais, ou seja, às folhas da árvore, apenas acompanhando a ordem que as ligações foram feitas. Por exemplo, a carga 158 possui ligação com a chave 198 e mais três outras chaves (não explicitadas na Figura 7 (a)); outro, é a carga 190, indicando ser uma carga terminal do sistema.

Uma característica importante a ser observada é que a representação das ligações é feita apenas uma única vez, a fim de não se ter redundância de dados. Por esse motivo que as cargas posteriores não têm representação de ligações já feitas na estrutura de dados apresentada.

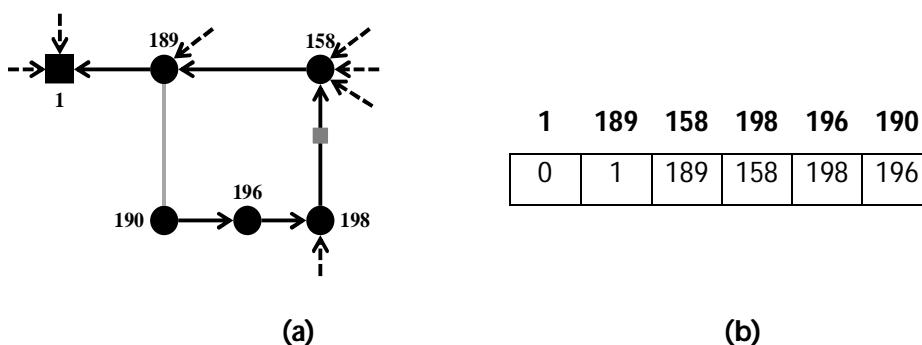
Outra característica importante e de grande interesse, que pode ser obtida desta estrutura de árvore geradora, é o número de cargas atendidas em uma

solução construída. Como apenas aquelas cargas atendidas terão ligação com a fonte de geração, apenas estas cargas farão parte desta estrutura. Assim, tendo conhecimento do número de cargas do sistema e o número de cargas atendidas, a conversão do percentual de atendimento torna-se fácil de ser obtido.

Diante do exposto, através da utilização desta estrutura, o algoritmo da ACO terá a capacidade de descobrir rapidamente os caminhos da raiz (fonte de geração) até as folhas (cargas terminais) desta árvore, através de um método que faça esta pesquisa, o chamado percurso em largura na árvore. Assim, esta estrutura será elementar durante a construção de uma solução para uma formiga, indicando se as soluções elaboradas satisfazem a condição de AG.

### 5.4.2 Árvore Reversa

Outra estrutura importante que auxiliará na construção de uma solução pelo algoritmo da ACO é a chamada Árvore Reversa (TOSO, 2006). A Árvore Reversa constitui uma estrutura muito similar à Árvore Geradora. Porém, ao invés de indicar um caminho da raiz às folhas de uma árvore, trabalha no sentido oposto, partindo de uma das folhas (ou outro ponto qualquer), visando chegar à raiz desta estrutura de árvore. De uma maneira mais formal, se cada barra é identificada por um valor inteiro no intervalo  $[1..n]$ , então  $v[i]$ ,  $i \in \{1, \dots, n\}$  armazena o inteiro que identifica o pai de  $i$  na Árvore Geradora. Para um melhor entendimento, a Figura 8 ilustra esta estrutura.



**Figura 8 – Representação da Árvore Reversa para o SEP, no qual (a) mostra sua representação gráfica e (b) sua representação estrutural.**

Pela Figura 8, são obtidas as seguintes características: a Figura 8 (a) mostra, graficamente, a Árvore Reversa do SEP da Figura 7 (a); pela Figura 8 (b), é obtida esta mesma árvore, porém segue-se a estrutura utilizada pelo algoritmo. Nesse caso, cada carga irá ter apenas uma carga “pai”, ou seja, não existindo ciclo neste sistema, cada linha deste será ligada em formato sequencial, fazendo com que a carga posterior da sequência tenha apenas uma carga que a anteceda, na qual terá ligação. Novamente, aquela carga que tiver uma representação de ligação de carga pai igual ao número zero (0) indicará uma característica diferente das demais, correspondendo, neste caso, à raiz da árvore, ou seja, a fonte de geração. Como exemplo, saindo da carga de número 196, todas as posições do vetor serão percorridas até que se chegue à primeira posição, que indica ser a raiz da Árvore Reversa.

Com isso, a representação de uma estrutura de Árvore Reversa torna-se simples e de fácil entendimento, podendo representá-la utilizando apenas um vetor, no qual o número de posições corresponde ao número de cargas do sistema.

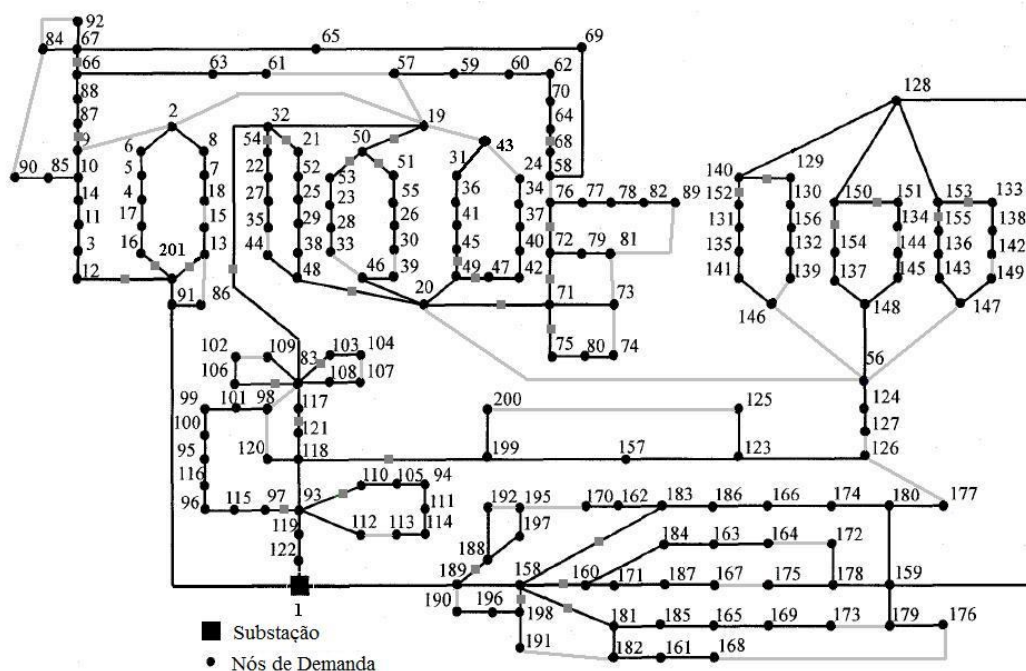
Por fim, a utilização desta estrutura será importante na ocorrência de uma falha, pois permite ao algoritmo percorrer, de forma eficiente, as ligações das soluções construídas até a fonte de geração. Desse modo, durante este percurso, é possível verificar se alguma linha da solução teve sobrecarga, o que seria necessário para a escolha de outro caminho alternativo pelas formigas do algoritmo.

## 5.5 Configurações do Sistema

O sistema desenvolvido utilizar-se-á de 2 configurações principais para o seu funcionamento: a primeira aborda a instância do SEP a ser utilizada; e a segunda, as configurações básicas adotadas para o algoritmo da ACO, através da estratégia *MMAS*. As demais configurações adotadas, como a estrutura de *hardware* e as configurações para a execução do algoritmo idealizado também são explicitadas.

## 5.5.1 SEP

Para trabalhar o problema abordado neste trabalho, será utilizada uma instância de um SEP, descrita por Ramírez-Rosado e Bernal-Agustín, (1998), denominada "*Distribution\_System\_01*", que é composta por uma unidade geradora, 201 cargas consumidoras, 39 linhas alternativas (chaves NO) e 37 switches (chaves NC). A ilustração deste sistema pode ser observada na Figura 9.



**Figura 9 – Instância "*Distribution\_System\_01*".**

A interpretação da Figura 9 dá-se da seguinte maneira: a unidade geradora é a parte principal do sistema, no qual fará a distribuição da energia para todas as demais cargas, por meio de uma linha conectora. Neste caso, ela é representada pelo elemento que corresponde ao número 1. Existem dois tipos de linhas de transmissão: as que estão em uso, representadas pelas linhas de cor preta entre as cargas; e as linhas que servem como caminho alternativo, que não estão em uso e estão representadas pela cor cinza. O motivo da não utilização das linhas alternativas já foi explicado, sendo que são linhas utilizadas para que o SEP mantenha suas características principais (Seção 3.3) no momento atual e seja possível mantê-las após algum restabelecimento decorrente de uma falha. Tais

linhas têm este aspecto por possuírem uma chave NO entre suas cargas extremas. Por fim, existem as linhas de transmissão em uso que possuem uma chave NC entre as cargas de suas extremidades, representadas pelas linhas com um quadrado cinza no meio desta ligação, como, por exemplo, a linha entre as cargas 188 e 189. São estas ligações que o algoritmo poderá alterar o estado das chaves, no caso de uma sobrecarga durante o restabelecimento.

Uma característica importante a respeito da forma de resolução proposta, e que talvez pareça uma limitação da implementação, é em relação às falhas que serão inseridas no sistema, para que o algoritmo da ACO possa trabalhar o problema a fim de encontrar a solução de um restabelecimento. Cada vez que o algoritmo executa sobre o problema, existe uma única falha no sistema. Assim, aqui não está sendo tratado o caso em que mais de uma falha ocorra simultaneamente no sistema. Caso isso seja necessário, a possibilidade seria de proceder com restabelecimentos sequenciais, ou seja, restabelecer cada falha isoladamente, como se esta fosse a única no sistema. Porém, isso não seria indicado para se obter um desempenho satisfatório.

### 5.5.2 *MMAS*

O algoritmo da ACO também terá suas configurações básicas que serão utilizadas durante sua execução. Primeiramente, como já dito, a estratégia utilizada para esta meta-heurística será uma das mais bem-sucedidas já desenvolvidas, a chamada *Max-Min Ant System*.

Em relação à configuração paramétrica, alguns parâmetros terão seus valores fixos, por definição. Já um outro parâmetro, o *número de formigas*, terá uma variação de seus valores para verificar a influência desta nos resultados.

Como parâmetros fixos, serão adotados a influência da trilha de feromônio ( $\alpha$ ), a influência da informação heurística ( $\beta$ ), o número de iterações, os níveis de feromônio máximo ( $\tau_{\max}$ ) e mínimo ( $\tau_{\min}$ ) e o máximo de tentativas. Tais parâmetros estão indicados na Tabela 1.

**Tabela 1 – Valores fixos estabelecidos para a ACO**

| Parâmetro           | Valor |
|---------------------|-------|
| $\alpha$            | 2     |
| $\beta$             | 0.1   |
| $\rho$              | 0.3   |
| $\tau_{\text{máx}}$ | 0.999 |
| $\tau_{\text{mín}}$ | 0.001 |
| Máx. Tentativas     | 1     |
| Núm. Iterações      | 50    |

É importante observar que todos esses parâmetros foram obtidos por intermédio de um processo de ajuste paramétrico com base em testes estatísticos sobre a falha “1 – 122”, em que se fez a análise dos resultados obtidos. Esse processo está descrito no Apêndice I.

Para o parâmetro *número de formigas*, será utilizada uma variação entre seus valores no desenvolver dos testes, como foi dito. Isso foi feito para verificar a influência deste parâmetro sobre o desempenho do algoritmo, pelo motivo de estar estritamente ligado ao processo de construção de uma solução pelo algoritmo. Os valores assumidos por esse parâmetro podem ser vistos na Tabela 2.

**Tabela 2 – Valores variáveis estabelecidos para a ACO**

| Parâmetro          | Valores assumidos |
|--------------------|-------------------|
| Número de Formigas | 5                 |
|                    | 15                |
|                    | 25                |
|                    | 30                |
|                    | 45                |
|                    | 60                |
|                    | 80                |
|                    | 100               |
| 200                |                   |

Em seguida, no Capítulo 6, serão mostrados os testes para as variações destes parâmetros, analisando qual o valor que mais se adapta aos parâmetros fixos assumidos.

### 5.5.3 Demais Configurações

Por fim, uma característica indispensável para a resolução do problema de restabelecimento dos SEPs são as configurações de *hardware* e de *software* utilizadas. Isto é imprescindível para futuras comparações com outras técnicas que venham a ser desenvolvidas para a resolução deste problema.

Para as configurações de hardware que foi executado o experimento, foi utilizado um computador com processador *Core 2 Duo* de 2GHz, com RAM de 2GB.

O algoritmo foi implementado em linguagem C#, sendo executado um total de 30 testes para cada situação de falha, que serão mostradas posteriormente.

## 5.6 Considerações Finais

Este capítulo buscou tratar a arquitetura que o sistema foi desenvolvido. Para isso, foi explicitado a forma que o problema foi implementado para o modelo de feromônios da ACO, as estruturas de dados adicionais utilizadas e as configurações paramétricas adotadas.

Na modelagem do problema para o modelo de feromônios da ACO, foi mostrada a forma que as formigas iriam construir suas soluções para o problema elétrico, seguindo uma determinada ordem para isso. Outro ponto foi a forma do depósito do feromônio para o problema, que está relacionada às chaves deste sistema. A informação heurística também foi essencial, pois absorve do problema decisões importantes a serem tomadas no início do processo de construção de soluções.

Em relação às estruturas de dados como um adicional ao algoritmo, a utilização da *Árvore Geradora*, para encontrar com rapidez o número de cargas atendidas, e da *Árvore Reversa*, para descobrir se determinada carga possui energia, são fundamentais para um bom desempenho do algoritmo.

E, finalmente, as configurações adotadas tanto pelo SEP, com o uso da instância "*Distribution\_System\_01*", quanto pela ACO, com a utilização de uma das mais promissoras de suas estratégias, a *MMAS*, e dos ajustes paramétricos, para os valores fixos e variantes, foram explicitados, além das demais configurações para a realização dos testes pelo algoritmo.

Assim, adotando tais configurações e condições estabelecidas, esses parâmetros serão a base com que o algoritmo idealizado será testado sobre o problema do SEP descrito.



# 6

## Resultados

*Este capítulo apresenta os resultados experimentais da arquitetura proposta, utilizando, para isso, todos os modelos, estruturas e configurações paramétricas apresentadas durante o Capítulo 5. Assim, será mais evidente o real benefício proporcionado pela estratégia idealizada, permitindo inferir os aspectos mais relevantes da técnica proposta.*

### 6.1 Considerações Iniciais

Este capítulo irá abordar os resultados experimentais a partir dos parâmetros estabelecidos anteriormente. Uma comparação entre resultados da literatura e os resultados obtidos pelo modelo idealizado também será conduzida.

Contudo, antes de iniciar qualquer execução do algoritmo proposto, foi analisada a influência que o parâmetro *número de formigas* teve sobre a busca de uma solução no presente problema de restabelecimento de SEPs.

Dessa forma, o tratamento dos resultados do algoritmo da ACO foi separado em duas seções distintas, na qual a Seção 6.2 aborda os testes para o parâmetro *número de formigas*, e a Seção 6.3 trata, efetivamente, os resultados do algoritmo da ACO com as configurações paramétricas estabelecidas, o que facilita e evidencia tal análise. Por fim, a Seção 6.4 traz os resultados mais recentes da literatura para a mesma instância do SEP utilizada, o que permite inferir algumas comparações entre as abordagens adotadas.

## 6.2 Experimento para o Parâmetro *Número de Formigas*

Adotando os parâmetros ora estabelecidos, a inserção de uma falha na linha "1 – 91" foi testada, seguindo a variação do parâmetro *número de formigas*. Note que a situação de falha para este teste foi em uma linha do sistema diferente da configuração dos demais parâmetros. O motivo de utilizar uma situação de falha diferente é pelo motivo de complexidade para a busca de soluções nesta situação, pois esta foi a falha que apresentou a otimização mais demorada e instável (número de alterações de chaves diferentes entre soluções) ao adotar alguns testes preliminares. Logo, o ajuste do parâmetro *número de formigas* para outra falha poderia prejudicar esta em sua otimização, gerando casos de possíveis não restabelecimentos. Assim, um ajuste de qualidade para esta falha irá, conseqüentemente, gerar um ajuste adequado para as demais situações de falhas, pelo motivo de terem apresentado uma necessidade de ajuste fino menor deste parâmetro, para encontrar a solução ótima.

Desta forma, para este teste foram realizadas 30 execuções para cada valor que o respectivo parâmetro assumia. Os resultados deste podem ser observados na Tabela 3.

A interpretação da Tabela 3 é feita da seguinte maneira: a primeira coluna indica o valor assumido pelo parâmetro *número de formigas* durante a sua execução; a segunda coluna mostra a porcentagem das vezes que o algoritmo, assumindo o respectivo *número de formigas*, encontrou qualquer solução válida; a terceira coluna indica a porcentagem de vezes que este encontrou a solução ótima para a falha; a quarta coluna indica o tempo médio de execução para encontrar uma solução válida; a quinta coluna indica o melhor tempo para encontrar uma solução; e, por fim, a última coluna indica o desvio padrão do algoritmo, para encontrar uma solução válida.

**Tabela 3 – Resultado para variação do parâmetro *número de formigas***

| Número de Formigas | Encontrou Solução (%) | Solução Ótima (%) | Tempo Médio (seg) | Melhor Tempo (seg) | Desvio Padrão (seg) |
|--------------------|-----------------------|-------------------|-------------------|--------------------|---------------------|
| 5                  | 30                    | 20                | 1.399             | 0.422              | 1.008               |
| 15                 | 43.33                 | 40                | 4.487             | 1.625              | 3.012               |
| 25                 | 10                    | 6.66              | 10.120            | 5.453              | 4.108               |
| 30                 | 100                   | 90                | 3.649             | 1.75               | 0.997               |
| 45                 | 100                   | 93.33             | 4.295             | 1.906              | 1.655               |
| 60                 | 100                   | 100               | 5.595             | 2.578              | 1.879               |
| 80                 | 100                   | 96.66             | 7.377             | 3.453              | 2.388               |
| 100                | 100                   | 100               | 6.309             | 4.281              | 1.529               |
| 200                | 100                   | 100               | 10.729            | 8.531              | 1.744               |

Analisando a Tabela 3, deve-se estabelecer qual valor o parâmetro *número de formigas* irá assumir para o experimento abrangendo outras falhas no SEP.

A característica mais importante, inicialmente, seria o tempo médio de execução para que o algoritmo da ACO encontrasse uma solução válida. Neste intuito, este parâmetro deveria assumir o valor 5, pois nos fornece o melhor tempo médio para encontrar uma solução válida. Contudo, esta escolha não seria eficiente, pois, para este parâmetro, o algoritmo somente encontrou soluções em 30% dos casos. Logo, isso quer dizer que em 70% das vezes, o algoritmo terminava sua execução com nenhuma resposta factível. Assim, mesmo que este tenha sido o melhor tempo médio, as vezes que foi executado sem obter uma resolução para o problema acaba tornando-o inadmissível para ser utilizado em um problema que exige uma resposta certa e rápida. Seguindo este raciocínio, os parâmetros 15 e 25 também foram descartados.

Os demais parâmetros são passíveis de serem utilizados. Somente resta, agora, analisar aquelas características alcançadas por tais parâmetros que sejam as mais promissoras.

Uma característica importante é que sempre seja encontrada a solução ótima para um problema. Isto, para um SEP, seria, na maioria das vezes, indispensável, pois significaria em menos gastos necessários para o restabelecimento. Contudo, o valor de tempo médio de execução assume a mesma importância, pois indica o tempo em que cargas do sistema ficarão sem energia; talvez, encontrar uma solução não tão qualificada seria menos prejudicial em questões de gastos a deixar certas cargas do sistema sem energia. Com isso, assumindo que a melhor escolha seria

minimizar o tempo que as cargas ficariam sem energia, e que encontrar a solução ótima na maioria dos casos é importante, o valor assumido pelo parâmetro *número de formigas* será igual a 30, pois além deste ter o melhor tempo médio, possui também o melhor tempo individual obtido e o menor desvio padrão em relação aos demais valores deste parâmetro, o que aumenta o grau de confiabilidade na solução encontrada. É importante salientar que, apesar de não conseguir encontrar a solução ótima em todos os casos, o valor obtido de 90% das vezes para encontrar esta solução é considerado elevado, o que não o desqualifica em relação aos demais resultados que conseguiram um percentual maior.

Como uma última interpretação destes resultados obtidos, é importante destacar a seguinte característica: à medida que se aumenta o número de formigas (para os valores deste parâmetro que encontraram soluções válidas em todos os casos), aumenta-se também o tempo médio, o melhor tempo individual e o desvio padrão obtidos, na maioria das vezes. Isso significa que o número de formigas utilizadas para a realização de um experimento é importante até certo ponto, pois, ao extrapolar este limite, o tempo gasto para calcular a construção de uma solução para cada formiga torna-se mais determinante que o benefício de se ter uma maior variação do número de soluções construídas ao final de uma iteração. Logo, este *overhead* de número de formigas por número de soluções variadas a cada iteração é uma característica importante a se considerar nestes casos.

Por fim, a escolha do valor 30 para o parâmetro *número de formigas* segue muito próximo a conclusão aferida por Dorigo e Stützle (2002) em relação a este parâmetro, que diz que um número interessante para este gira em torno do número de vértices do problema. Como a instância "*Distribution\_System\_01*" possui 39 chaves NO e 37 chaves NC, o valor escolhido foi o que mais se aproximou desta premissa, indicando que também, para o problema de restabelecimento de SEPs, esta é uma escolha promissora para se obter resultados satisfatórios.

Assim, adotando esta configuração do parâmetro *número de formigas*, todos os parâmetros da ACO estão determinados, sendo possível, então, proceder com os testes sobre a instância indicada.

## 6.3 Experimento com os Parâmetros Estabelecidos

Findada a fase de configuração dos parâmetros do algoritmo da ACO, serão mostrados os resultados encontrados para os restabelecimentos feitos, seguindo vários casos de falha embutidos no SEP descrito.

Seguindo a característica de número de aberturas e fechamentos de chaves realizadas para proceder-se com um restabelecimento, os resultados foram divididos em três partes distintas, sendo a primeira (Tabela 4) os casos de falha que necessitaram de mais de um fechamento de uma chave NO para o restabelecimento; a segunda (Tabela 5), os casos em que somente um fechamento de chave NO foi necessário, porém várias soluções eram possíveis; e a terceira (Tabela 6), os casos em que somente um fechamento de chave NO foi necessário, sendo este a única opção para o restabelecimento.

Para as Tabelas de 4 a 6, a interpretação dar-se-á da mesma maneira: para a primeira coluna denominada "*Exemplos*", têm-se as informações de uma situação de falha na linha entre as respectivas cargas (*Linha com Falha*), indicando o percentual de cargas atendidas mesmo com tal situação de falha (*P%*). A segunda coluna, com o nome de "*Solução encontrada*" mostra os dados a respeito da solução obtida pelo algoritmo da ACO, contendo as chaves normalmente abertas que foram fechadas (*NO*), as chaves normalmente fechadas que foram abertas (*NC*) e o respectivo melhor tempo para esta operação (*t(s)*). Por fim, a última coluna, chamada de "*Análise Estatística*", mostra alguns dados estatísticos acerca dos restabelecimentos que tiveram o mesmo número de aberturas e fechamentos de chaves, para a sua respectiva situação de falha, em que são indicados o tempo médio para encontrar uma solução (*Tempo Médio*), o melhor tempo desenvolvido pelo algoritmo para realizar o restabelecimento (*Melhor Tempo*) e o desvio padrão dos resultados encontrados (*Desvio Padrão*).

**Tabela 4 – Resultados de simulações de falhas – Número de fechamentos de chaves NO maior que 1**

| <i>Exemplos</i> |       | <i>Solução Encontrada</i>           |                        |       | <i>Análise Estatística</i> |              |               |
|-----------------|-------|-------------------------------------|------------------------|-------|----------------------------|--------------|---------------|
| Linha com Falha | P%    | NO                                  | NC                     | t(s)  | Tempo Médio                | Melhor Tempo | Desvio Padrão |
| 1 – 122         | 54.51 | (2 – 19)<br>(126 – 177)             | 117 – 121              | 1.677 | 1.290                      | 0.394        | 0.639         |
|                 |       | (2 – 19)<br>(126 – 127)             | 117 – 121              | 0.884 |                            |              |               |
|                 |       | (20 – 56)<br>(2 – 19)               | 20 – 48                | 1.270 |                            |              |               |
|                 |       | (19 – 57)<br>(126 – 127)            | 117 – 121              | 0.394 |                            |              |               |
|                 |       | (58 – 76)<br>(126 – 127)            | 117 – 121              | 0.396 |                            |              |               |
|                 |       | (126 – 177)<br>(19 – 57)            | 117 – 121              | 2.428 |                            |              |               |
|                 |       | (126 – 177)<br>(58 – 76)            | 117 – 121              | 0.789 |                            |              |               |
| 1 – 189         | 62.87 | (126 – 127)<br>(58 – 76)            | 117 – 121              | 0.389 | 1.373                      | 0.389        | 0.526         |
|                 |       | (126 – 177)<br>(58 – 76)            | 117 – 121              | 0.406 |                            |              |               |
| 1 – 91          | 82.62 | (2 – 19)<br>(20 – 56)               | 20 – 48                | 1.726 | 4.353                      | 1.726        | 2.145         |
|                 |       | (19 – 57)<br>(20 – 56)              | 20 – 48                | 2.898 |                            |              |               |
|                 |       | (2 – 19)<br>(20 – 56)<br>(58 – 76)  | (20 – 71)<br>(21 – 32) | 3.023 | 4.531                      | 2.612        | 2.698         |
|                 |       | (19 – 57)<br>(20 – 56)<br>(58 – 76) | (20 – 71)<br>(21 – 32) | 2.612 |                            |              |               |

Na Tabela 4, são mostradas as falhas que geram um maior número de cargas sem energia. Logo, devido ao seu maior número de possibilidades para proceder com o restabelecimento, tais falhas apresentam também os maiores tempos para encontrar uma solução válida.

Uma característica da Tabela 4 está em torno da falha “1 – 91”. Nesta falha, a otimização pelo algoritmo da ACO foi a mais difícil de obter um desempenho satisfatório, tendo até mesmo encontrado soluções não tão qualificadas para o

problema em alguns casos (cinco alterações de chaves), como seria de se esperar. Logo, este é o motivo pela separação dos dados estatísticos para esta falha.

**Tabela 5 – Resultados de simulações de falhas – Número de fechamento de chaves NO igual a 1, apresentando várias soluções**

| <i>Exemplos</i> |       | <i>Solução Encontrada</i> |    |       | <i>Análise Estatística</i> |              |               |
|-----------------|-------|---------------------------|----|-------|----------------------------|--------------|---------------|
| Linha com Falha | P%    | NO                        | NC | t(s)  | Tempo Médio                | Melhor Tempo | Desvio Padrão |
| 9 – 10          | 89.99 | 2 – 9                     | –  | 0.179 | 0.235                      | 0.179        | 0.037         |
|                 |       | 84 – 90                   | –  | 0.197 |                            |              |               |
| 13 – 201        | 99.51 | 15 – 18                   | –  | 0.046 | 0.048                      | 0.046        | 0.003         |
|                 |       | 13 – 86                   | –  | 0.046 |                            |              |               |
| 19 – 32         | 93.91 | 2 – 19                    | –  | 0.047 | 0.047                      | 0.046        | 0.000         |
|                 |       | 19 – 43                   | –  | 0.048 |                            |              |               |
|                 |       | 19 – 57                   | –  | 0.046 |                            |              |               |
|                 |       | 30 – 39                   | –  | 0.046 |                            |              |               |
|                 |       | 33 – 46                   | –  | 0.046 |                            |              |               |
| 56 – 148        | 99.19 | 20 – 56                   | –  | 0.046 | 0.047                      | 0.046        | 0.001         |
|                 |       | 56 – 146                  | –  | 0.046 |                            |              |               |
|                 |       | 56 – 147                  | –  | 0.047 |                            |              |               |
|                 |       | 126 – 127                 | –  | 0.046 |                            |              |               |
| 93 – 118        | 62.98 | 83 – 98                   | –  | 0.268 | 0.337                      | 0.233        | 0.054         |
|                 |       | 98 – 120                  | –  | 0.233 |                            |              |               |
| 12 – 201        | 86.75 | 2 – 9                     | –  | 0.205 | 0.294                      | 0.205        | 0.041         |

A Tabela 5 mostra um conjunto de falhas e soluções mais simplificadas, apresentando várias soluções em uma única alteração de chave NO. Nos resultados obtidos, pode ser observado que quanto menor a região desprovida de energia, decorrente da falha imposta ao sistema, maior a rapidez para a sua resolução. Outra característica a ser observada na Tabela 5 é que apesar de a falha “12 – 201” apresentar uma única solução de restabelecimento nos resultados obtidos, esta possui outras situações de restabelecimento possíveis, porém, não sendo a solução ótima; diante disso, o algoritmo da ACO conseguiu encontrar somente a solução ótima para este caso, por não ser uma falha difícil de ser restabelecida.

**Tabela 6 – Resultados de simulações de falhas – Número de fechamento de chaves NO igual a 1, apresentando única solução**

| <i>Exemplos</i> |       | <i>Solução Encontrada</i> |    |       | <i>Análise Estatística</i> |              |               |
|-----------------|-------|---------------------------|----|-------|----------------------------|--------------|---------------|
| Linha com Falha | P%    | NO                        | NC | t(s)  | Tempo Médio                | Melhor Tempo | Desvio Padrão |
| 76 – 77         | 98.16 | 81 – 89                   | –  | 0.046 | 0.047                      | 0.046        | 0.001         |
| 93 – 110        | 95.68 | 112 – 113                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 128 – 140       | 93.62 | 56 – 146                  | –  | 0.046 | 0.047                      | 0.046        | 0.001         |
| 160 – 184       | 98.32 | 164 – 172                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 2 – 8           | 99.18 | 15 – 18                   | –  | 0.045 | 0.046                      | 0.045        | 0.001         |
| 10 – 85         | 99.10 | 84 – 90                   | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 32 – 54         | 98.24 | 35 – 44                   | –  | 0.046 | 0.047                      | 0.046        | 0.001         |
| 40 – 42         | 97.12 | 24 – 43                   | –  | 0.046 | 0.047                      | 0.046        | 0.002         |
| 50 – 51         | 96.89 | 30 – 39                   | –  | 0.046 | 0.047                      | 0.046        | 0.002         |
| 50 – 53         | 97.99 | 33 – 46                   | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 63 – 66         | 98.91 | 57 – 61                   | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 71 – 75         | 98.22 | 73 – 74                   | –  | 0.046 | 0.047                      | 0.046        | 0.001         |
| 78 – 82         | 99.08 | 81 – 89                   | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 83 – 106        | 99.08 | 102 – 109                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 83 – 108        | 99.57 | 104 – 107                 | –  | 0.046 | 0.046                      | 0.046        | 0.000         |
| 111 – 114       | 98.91 | 112 – 113                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 132 – 156       | 98.85 | 139 – 146                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 138 – 142       | 97.69 | 142 – 149                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 145 – 148       | 99.16 | 134 – 144                 | –  | 0.045 | 0.046                      | 0.045        | 0.001         |
| 160 – 171       | 98.58 | 167 – 175                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |
| 161 – 168       | 99.85 | 168 – 176                 | –  | 0.046 | 0.046                      | 0.046        | 0.001         |

Por fim, a Tabela 6 mostra os resultados obtidos para as situações mais “simples” de falhas no sistema. Para todos os casos, somente a solução encontrada era a solução possível, pois eram as únicas chaves vizinhas à região de falha. Assim, o trabalho feito pelo algoritmo da ACO é bastante simplificado devido a estratégia de escolha de chaves NO: como a heurística para a escolha destas chaves está relacionada às chaves vizinhas à região de falha, as únicas chaves disponíveis após os cálculos da lista de chaves vizinhas acabava sendo as próprias chaves que eram as soluções para o problema. Assim, era necessário somente o tempo para calcular tal lista de chaves para que o algoritmo pudesse fazer a escolha por uma destas que são a solução para o problema.

Diante disso, pode-se observar que a utilização de estruturas de dados adicionais a uma técnica de otimização acaba gerando um ganho de desempenho



ao algoritmo, sendo fortemente indicado o seu uso sempre que forem passíveis de implementação.

## 6.4 Comparação com Outras Abordagens

Adotando com referencial o trabalho de Lambert-Torres *et al.* (2009), que utiliza a mesma instância do SEP, serão mostrados os seus resultados obtidos pelas estratégias dos Algoritmos Genéticos e da Otimização por Enxame de Partículas. Contudo, as análises inferidas para comparação com estas abordagens serão somente superficiais, pelo motivo de que estes autores apenas disponibilizaram os seus melhores resultados, sem indicarem as configurações da máquina que foi executado o seu experimento, sequer qualquer dado estatístico acerca de seus resultados.

Os resultados, para uma melhor visualização, foram divididos novamente em três partes, seguindo a mesma ideia dos resultados anteriores. Logo, a interpretação das Tabelas de 7 a 9 segue, basicamente, os mesmo princípios, sendo que para a segunda coluna, nomeada "*Situação de Restabelecimento com a PSO*" mostra a sequência de chaves NO e NC, seguido do tempo para essa resolução, para o algoritmo da PSO; e a última coluna, chamada de "*Situação de Restabelecimento com o GA*" evidencia também estas características, contudo para o Algoritmo Genético utilizado no experimento.

Em seguida, são mostrados tais resultados, sendo feita uma breve comparação com os resultados obtidos com o modelo da ACO.

**Tabela 7 – Resultados de outras abordagens para falhas com várias alterações de chaves**

| <i>Exemplos</i> |       | <i>Situação de Restabelecimento com a PSO</i> |           |       | <i>Situação de Restabelecimento com o GA</i> |                        |       |
|-----------------|-------|---|-----------|-------|--|------------------------|-------|
| Linha com Falha | P%    | NO  | NC        | t(s)  | NO   | NC                     | t(s)  |
| 1 - 122         | 54.51 | (2 - 19)<br>(126 - 177)                       | 117 - 121 | 38.06 | (20 - 56)<br>(2 - 19)                        | 48-20                  | 68.48 |
|                 |       | (19 - 57)<br>(126 - 127)                      | 117 - 121 | 43.37 |  |                        |       |
|                 |       | (58 - 76)<br>(126 - 177)                      | 117 - 121 | 34.05 | (19 - 57)<br>(126 - 127)                     | 117-121                |       |
|                 |       | (2 - 19)<br>(126 - 127)                       | 117 - 121 | 33.53 |  |                        |       |
| 1 - 189         | 62.87 | (58 - 76)<br>(126 - 127)                      | 117 - 121 | 34.36 | (126 - 127)<br>(126 - 177)                   | 56-124                 | 69.90 |
| 1-91            | 82.62 | (2 - 19)<br>(20 - 56)                         | 20 - 48   | 44.19 | (19 - 57)<br>(76 - 58)<br>(2 - 9)            | (70 - 62)<br>(67 - 65) | 75.10 |
|                 |       |   |           |       | (19 - 57)<br>(2 - 9)                         | 58-68                  |       |

Os resultados obtidos na Tabela 7, alcançados pela PSO e pelo GA, foram os mais significantes em comparação aos resultados da ACO, sendo em alguns casos em torno de 1 minuto de diferença no tempo para encontrar uma solução. Além disso, como pode ser observado nos resultados da abordagem que utiliza o GA, esta estratégia teve a desvantagem de encontrar soluções não tão qualificadas quanto à PSO e à ACO. Assim, os resultados obtidos pelo GA foram os mais instáveis, em relação às outras abordagens.

**Tabela 8 – Resultados de outras abordagens para falhas com várias soluções, sendo possível uma única alteração de chave**

| <i>Exemplos</i>        |           | <i>Situação de Restabelecimento com a PSO</i> |           |             | <i>Situação de Restabelecimento com o GA</i> |                          |             |
|------------------------|-----------|---|-----------|-------------|--|--------------------------|-------------|
| <i>Linha com Falha</i> | <i>P%</i> | <b>NO</b>                                     | <b>NC</b> | <b>t(s)</b> | <b>NO</b>                                    | <b>NC</b>                | <b>t(s)</b> |
| 9 - 10                 | 89.99     | 84 - 90                                       | -         | 0.22        | (76 - 58)<br>(19 - 57)<br>(90 - 84)          | (59 - 60)<br>(20 - 71)   | 61.08       |
|                        |           | 2 - 9   | -         | 0.35        | (2 - 9)<br>(19 - 57)                         | 62-60                    |             |
| 13 - 201               | 99.51     | 15 - 18                                       | -         | 0.16        | (15 - 18)                                    | 2-8                      | 57.45       |
|                        |           | 13 - 86                                       | -         | 0.19        | (13 - 86)<br>13 - 86                         | -                        |             |
| 19 - 32                | 93.91     | 33 - 46                                       | -         | 0.17        | (33 - 46)<br>(30 - 39)                       | 50-51                    | 59.14       |
|                        |           | 19 - 57                                       | -         | 0.19        | (19 - 57)                                    | (49 - 45)<br>(64 - 70)   |             |
|                        |           | 2 - 19  | -         | 0.16        | (19 - 43)                                    |                          |             |
|                        |           | 19 - 43                                       | -         | 0.18        | (2 - 19)                                     |                          |             |
| 56 - 148               | 99.19     | 20 - 56                                       | -         | 0.16        | (126 - 127)                                  | 38-48                    | 61.01       |
|                        |           | 126 - 127                                     | -         | 0.18        | (20 - 56)                                    |                          |             |
| 93 - 118               | 62.98     | 56 - 147                                      | -         | 0.16        | (147 - 56)<br>(20 - 56)<br>(126 - 127)       | (143 - 147)<br>(38 - 48) | 73.71       |
|                        |           | 83 - 98                                       | -         | 0.20        | (98 - 83)                                    | 62-60                    |             |
|                        |           | 98 - 120                                      | -         | 0.16        | (19 - 57)                                    |                          |             |
| 12 - 201               | 86.75     | 2 - 9   | -         | 0.31        | (19 - 57)<br>(2 - 9)                         | 62-60                    | 68.10       |
|                        |           |   |           |             | 2-9  |                          |             |

Os resultados obtidos e explicitados na Tabela 8 mostram uma diferença menos significativa dos resultados da ACO em relação à PSO. Porém, para o GA, continua com um tempo e qualidade dos resultados similares aos da Tabela 7. Novamente, os resultados obtidos pelo GA mostram-se bastante instáveis, se comparados às outras abordagens.

**Tabela 9 – Resultados de outras abordagens com única solução**

| <i>Exemplos</i> |       | <i>Situação de Restabelecimento com a PSO</i> |                 |      | <i>Situação de Restabelecimento com o GA</i> |                 |      |
|-----------------|-------|---|-----------------|------|--|-----------------|------|
| Linha com Falha | P%    | NO  | Linha com Falha | P%   | NO   | Linha com Falha | P%   |
| 76 - 77         | 98.16 | 81 - 89                                       | -               | 0.17 | 81 - 89                                      | -               | 0.10 |
| 93 - 110        | 95.68 | 112 - 113                                     | -               | 0.20 | 112 - 113                                    | -               | 0.10 |
| 128 - 140       | 93.62 | 56 - 146                                      | -               | 0.17 | 56 - 146                                     | -               | 0.10 |
| 160 - 184       | 98.32 | 164 - 172                                     | -               | 0.22 | 164 - 172                                    | -               | 0.10 |
| 2 - 8           | 99.18 | 15 - 18                                       | -               | 0.17 | 15 - 18                                      | -               | 0.10 |
| 10 - 85         | 99.10 | 84 - 90                                       | -               | 0.17 | 84-90  | -               | 0.10 |
| 32 - 54         | 98.24 | 35 - 44                                       | -               | 0.17 | 35-44  | -               | 0.10 |
| 40 - 42         | 97.12 | 24 - 43                                       | -               | 0.21 | 24 - 43                                      | -               | 0.10 |
| 50 - 51         | 96.89 | 30 - 39                                       | -               | 0.30 | 30 - 39                                      | -               | 0.10 |
| 50 - 53         | 97.99 | 33 - 46                                       | -               | 0.17 | 33 - 46                                      | -               | 0.10 |
| 63 - 66         | 98.91 | 57 - 61                                       | -               | 0.15 | 57-61  | -               | 0.10 |
| 71 - 75         | 98.22 | 73 - 74                                       | -               | 0.16 | 73 - 74                                      | -               | 0.10 |
| 78 - 82         | 99.08 | 81 - 89                                       | -               | 0.23 | 81 - 89                                      | -               | 0.10 |
| 83 - 106        | 99.08 | 102 - 109                                     | -               | 0.17 | 102-109                                      | -               | 0.10 |
| 83 - 108        | 99.57 | 104 - 107                                     | -               | 0.17 | 104-107                                      | -               | 0.10 |
| 111 - 114       | 98.91 | 112 - 113                                     | -               | 0.19 | 112-113                                      | -               | 0.10 |
| 132 - 156       | 98.85 | 139 - 146                                     | -               | 0.17 | 139 - 146                                    | -               | 0.10 |
| 138 - 142       | 97.69 | 142 - 149                                     | -               | 0.18 | 142 - 149                                    | -               | 0.10 |
| 145 - 148       | 99.16 | 134 - 144                                     | -               | 0.19 | 134 - 144                                    | -               | 0.10 |
| 160 - 171       | 98.58 | 167 - 175                                     | -               | 0.19 | 167 - 175                                    | -               | 0.10 |
| 161 - 168       | 99.85 | 168 - 176                                     | -               | 0.16 | 168 - 176                                    | -               | 0.10 |

Por fim, os resultados apresentados na Tabela 9 são os mais próximos dos resultados obtidos pela ACO. Contudo, a ACO ainda sim mostra uma significativa diferença para a busca da melhor solução, em comparação as duas estratégias ali utilizadas.

Contudo, apesar de toda a discrepância obtida entre estes resultados, não é possível afirmar que tanto a PSO quanto a GA são más opções de escolha de técnicas de otimização para este tipo de problema, pelo motivo de não se ter a informação completa das configurações paramétricas e estruturas destes algoritmos utilizadas para seus experimentos, como já foi discutido. Porém, pela dificuldade do problema de otimização, pode ser afirmado que a ACO apresentou um eficiente resultado, fazendo com que ela seja uma estratégia fortemente indicada para resolução de problemas desta natureza.

## 6.5 Considerações Finais

Este capítulo buscou alcançar os objetivos iniciais do trabalho, ao explicitar os resultados obtidos com a técnica proposta.

Inicialmente, visando obter um desempenho satisfatório, o ajuste do parâmetro *número de formigas* foi feito para a ACO, conservando os parâmetros fixos adotados no capítulo anterior.

Após um ajuste fino de todos os parâmetros, a ACO foi aplicada a várias situações de falha na instância adotada, visando encontrar soluções para o problema.

Por fim, para evidenciar o real benefício da aplicação da técnica proposta, os resultados mais recentes da literatura para esta instância foram mostrados, permitindo uma comparação direta entre as abordagens desenvolvidas. Diante disso, fica claro os reais benefícios proporcionados decorrentes da adoção da ACO com uma heurística para resolução de problemas exponenciais de otimização combinatória.



# 7

## Conclusões

*Este capítulo apresenta as conclusões desta monografia. É organizado da seguinte forma: Na Seção 7.1, são apresentados o resumo e as contribuições deste trabalho. Na Seção 7.2, são apresentadas as futuras diretrizes que poderão ser seguidas a partir do que foi desenvolvido.*

### 7.1 Conclusões Gerais

Neste trabalho, uma alternativa para a resolução do problema de Restabelecimento de Sistemas Elétricos de Potência foi apresentada. Para isso, foi proposta a aplicação de uma técnica de inteligência artificial para otimização combinatória, denominada Otimização por Colônia de Formigas (ACO). Dessa forma, a ACO teve a tarefa de escolher as chaves e a ordem de manipulação destas após a ocorrência de uma falha, de modo a restabelecer todas as possíveis cargas do sistema, respeitando todas as restrições impostas neste problema.

Como contribuição deste trabalho, a implementação da ACO para a manipulação dos SEPs, após as falhas embutidas nestes, seguiu uma das estratégias mais promissoras desenvolvidas para esta meta-heurística, denominada *Max-Min Ant System*. Pelos resultados obtidos, a implementação desta técnica apresentou resultados eficientes se comparado ao problema exponencial original, confirmando a ACO com uma promissora alternativa quando aplicada em problemas combinatórios da área da Engenharia Elétrica.

Diante do exposto, pode-se dizer que a abordagem utilizada conseguiu atender os objetivos propostos neste trabalho. Em adicional, o tempo gasto e a qualidade das soluções obtidas mostram que a aplicação da ACO em problemas práticos da área da Engenharia Elétrica, que visam o restabelecimento de SEPs, trará benefícios importantes, visto que se obteve a redução do tempo gasto para realizar a tarefa de restabelecimento de uma forma factível e eficiente em um problema real.

## 7.2 Trabalhos Futuros

Como trabalhos futuros, a meta será testar o modelo implementado para outras instâncias de Sistemas Elétricos de Potência presentes na literatura. A comparação com outras estratégias da meta-heurística ACO também será elaborada, a fim de verificar a influência de cada uma neste problema. Com isso, um novo processo de ajuste paramétrico deverá ser realizado, configurando tais parâmetros para cada estratégia utilizada. Outro ponto considerado será a aplicação de novas formas de manipulação das árvores geradoras e árvores reversas, a fim de melhorar as verificações feitas por estas estruturas. Por fim, a adição da Lógica Paraconsistente (DA COSTA; HENSCHÉ; SUBRAHMANIN, 1990) ao modelo idealizado será feita, para auxiliar as formigas na tomada de decisão, visando melhorar a eficiência do modelo.



# 8

## Referências Bibliográficas

Blum C., Roli, A. & Dorigo, M. *HC-ACO: The hyper-cube framework for Ant Colony Optimization*. Proceedings of MIC'2001 – Metaheuristics International Conference, vol. 2, Porto, Portugal, 2001, pp. 399–403.

Bonabeau, E., Dorigo, M. & Théraulaz, G. *From Natural to Artificial Swarm Intelligence*. Oxford University Press, 1999.

Bullnheimer, B., Hartl, R. F. & Strauss, C. *A New Rank Based Version of the Ant System: A Computational Study*. Central European Journal for Operations Research and Economics, 7(1):25-38, 1999.

Cormen, T. H., Leiserson, C. E, Rivest, R. L & Stein, C. *Algoritmos: Teoria e Prática*. Editora Campus, 2ª Edição, 2002.

Da Costa, N. C. A., Hensche, L. J. & Subrahmanin, V. S. *Automatic Theorem Proving in Paraconsistent Logics: Theory and Implementation*. Estudos Avançados, Coleção Documentos n° 3, USP, SP, Brazil, 1990.

Dorigo, M. *Optimization, Learning and Natural Algorithms*. Ph.D. thesis, Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy, 1992.

Dorigo, M.; Birattari, M. & Stützle, T. *Ant Colony Optimization - Artificial Ants as a Computational Intelligence Technique*. Published by IRIDIA, Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle, Université Libre de Bruxelles, Belgium, September 2006.

- Dorigo, M., Caro, G. & Gambardella, L. M. *Ant Algorithms for Discrete Optimization*. Technical Report 98-10, IRIDIA, Université Libre de Bruxelles. Submitted to *Artificial Life*, 1998.
- Dorigo, M. & Gambardella, L. M. *Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem*. IEEE Transactions on Evolutionary Computation, 1(1):53-66, 1997.
- Dorigo, M., Maniezzo, V. & Colorni, A. *Ant System: Optimization by a colony of cooperating agents*. IEEE Transactions on Systems, Man, and Cybernetics-Part B, 26(1), 29-41, 1996.
- Dorigo, M. & Socha, K. *An Introduction to Ant Colony Optimization*. Published as a chapter in *Approximation Algorithms and Metaheuristics*. IRIDIA, Université Libre de Bruxelles, Belgium, April 30, 2007.
- Dorigo, M. & Stützle, T. *The Ant Colony Optimization Metaheuristic: Algorithms, Applications and Advances*. In F. Glover and G. Kochenberger, editors, *Handbook of Metaheuristics*, volume 57 of *International Series in Operations Research & Management Science*, chapter 9, pages 251-285. Kluwer Academic Publishers, Boston, MA, 2002.
- Glover, F. *Tabu search—Part I*, ORSAJ. Comput. 1 (3) (1989) 190–206.
- Glover, F. *Tabu search—Part II*, ORSAJ. Comput. 2 (1) (1990) 4–32.
- Jünger, M., Reinelt, G. & Rinaldi, G. *The Traveling Salesman Problem: a Bibliography*. M. Dellamico, F. Maffioli, S. Martello (eds.): *Annotated Bibliography in Combinatorial Optimization*, Willey, 1997, 199-221.
- Kennedy, J. & Eberhart, R. C. *Particle swarm optimization*. Proc. IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, pp. IV: 1942-1948, 1995.

- Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. *Optimization by simulated annealing*. Science 220 (4598) 671–680, 1983.
- Lambert-Torres, G., Martins, H. G., Coutinho, M. P., Moraes, C. H. V., SALOMON, C. P., Matsunaga, F. M. & Carminati, R. A. *Comparison between PSO and GA in System Restoration Solution*. In: 15th International Conference on Intelligent System Applications to Power Systems, ISAP 2009, Curitiba. Proceedings of the 15th International Conference on Intelligent System Applications to Power Systems. IEEE Press, 2009.
- Liu, Y. & Gu, X. *Reconfiguration of Network Skeleton Based on Discrete Particle-Swarm Optimization for Black-Start Restoration*. Proc. IEEE Power Engineering Society General Meeting, Montreal, Canada, 2006.
- Mansour, M. R. *Algoritmo para obtenção de planos de restabelecimento para sistemas de distribuição de grande porte*. São Carlos, 2009, 110p. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo.
- Ramírez-Rosado, I. J. & Bernal-Agustín, J. L. *Genetic Algorithms Applied to the Design of Large Power Distribution Systems*. IEEE Transactions Power Systems, Vol. 13, No. 2, 1998.
- Rossi, R. *Systemic Hierarchic Classifier for Electrical Nets High-Voltage*. Ph. D. Thesis, in Portuguese, Universidade Federal de Itajubá, Brasil, 2000.
- Stützle, T. & Hoos, H. H. *MAX-MIN Ant System*. Future Generation Computer Systems. 16(8):889-914, 2000.
- Toso, R. F., *Algoritmos para Atualização de Árvores Geradoras Mínimas em Grafos Dinâmicos*. Dissertação de Mestrado, Universidade Federal Fluminense, Niterói, 2006.

Watanabe, I. *An ACO Algorithm for Service Restoration in Power Distribution Systems*. In IEEE Congress on Evolutionary Computation, 2005. The 2005, pages 2864 – 2871, Vol. 3, 2005.

Zanetta Jr., L. C. *Fundamentos de Sistemas Elétricos de Potência*. Editora Livraria da Física, São Paulo, 2005.

# 9

## Apêndice I

*Esta seção apresenta a forma que os parâmetros do algoritmo da ACO foram ajustados para serem utilizados neste trabalho.*

### 9.1 Considerações Iniciais

A forma que um parâmetro está ajustado para um experimento é determinante para um desempenho satisfatório de qualquer algoritmo. Assim, o procedimento de ajuste paramétrico é algo que deve ser dado uma importância maior, devido ao impacto no resultado final de um experimento.

Dessa forma, a seguir é mostrado o processo de ajuste paramétrico dos parâmetros do algoritmo da ACO implementado neste trabalho.

### 9.2 Parâmetros

Os parâmetros a serem ajustados para o algoritmo da ACO foram:

- $\alpha$ : fator que indica a importância que a trilha de feromônio terá durante o cálculo de probabilidade de escolha da matriz de probabilidades;
- $\beta$ : fator que indica a importância que a informação heurística terá durante o cálculo de probabilidade de escolha da matriz de probabilidades;
- $\rho$ : fator que determina o nível de evaporação das trilhas de feromônio;
- $\tau_{\text{máx}}$ : limitante superior do nível de feromônio;

- $\tau_{\min}$ : limitante inferior do nível de feromônio;
- Máx. Tentativas: máximo de tentativas do algoritmo para encontrar uma solução. A cada tentativa, as trilhas de feromônio são reinicializadas;
- Núm. Iterações: máximo de iterações do algoritmo para encontrar uma solução. A cada iteração, as trilhas de feromônio são atualizadas de acordo com a respectiva trilha da iteração anterior.

### 9.3 Ajustes Paramétricos

Devido o grande número de parâmetros do algoritmo da ACO a serem ajustados, quatro destes parâmetros foram ajustados empiricamente, sendo estes:

- $\tau_{\max}$  e  $\tau_{\min}$ : foi definido que uma trilha de feromônio não terá influência total ou nenhuma influência durante o cálculo de probabilidades, fazendo com que estes parâmetros assumissem os valores *0.999* e *0.001*, respectivamente;
- Máx. Tentativas: como a pretensão era encontrar um menor tempo de execução, este parâmetro assumiu o valor *1*, pois não é interessante reinicializarmos as trilhas de feromônio durante o processo de busca de soluções devido ao tempo gasto para se obter um novo conhecimento para estas trilhas;
- Núm. Iterações: também pelo motivo de menor tempo, o número máximo de iterações foi definido em *50*.

Para os demais parâmetros, a escolha foi definida da seguinte forma:

- Primeiramente, foram testadas todas as combinações possíveis para os valores paramétricos da Tabela 10, no qual foram executados *30* experimentos para cada combinação feita;

- Em seguida, foi verificada a interação entre estes parâmetros, de acordo com o tempo médio de execução, para se obter as melhores configurações. Salienta-se que todas estas configurações tiveram a mesma qualidade de solução. A interação entre tais parâmetros pode ser vista da Figura 10 até a Figura 12, sendo que as sete melhores interações (Tabela 11) foram escolhidas para um novo experimento.
- Por fim, após a execução de 30 experimentos para cada combinação paramétrica da Tabela 11, foi escolhida a combinação paramétrica com o menor tempo médio de execução. O tempo médio de execução pode ser visto na Tabela 12.

**Tabela 10 – Valores paramétricos iniciais**

| Parâmetro | Valor |
|-----------|-------|
| $\alpha$  | 0.1   |
|           | 0.5   |
|           | 1.0   |
|           | 1.5   |
|           | 2.0   |
| $\beta$   | 0.1   |
|           | 0.5   |
|           | 1.0   |
|           | 1.5   |
|           | 2.0   |
| $\rho$    | 0.01  |
|           | 0.05  |
|           | 0.09  |
|           | 0.2   |
|           | 0.3   |

**Tabela 11 – Melhores valores paramétricos**

| $\alpha$ | $\beta$ | $\rho$ |
|----------|---------|--------|
| 0.5      | 0.1     | 0.09   |
| 1.5      | 0.1     | 0.05   |
| 1.5      | 0.5     | 0.2    |
| 2.0      | 0.1     | 0.2    |
| 2.0      | 0.1     | 0.3    |
| 2.0      | 0.5     | 0.09   |
| 2.0      | 0.5     | 0.3    |

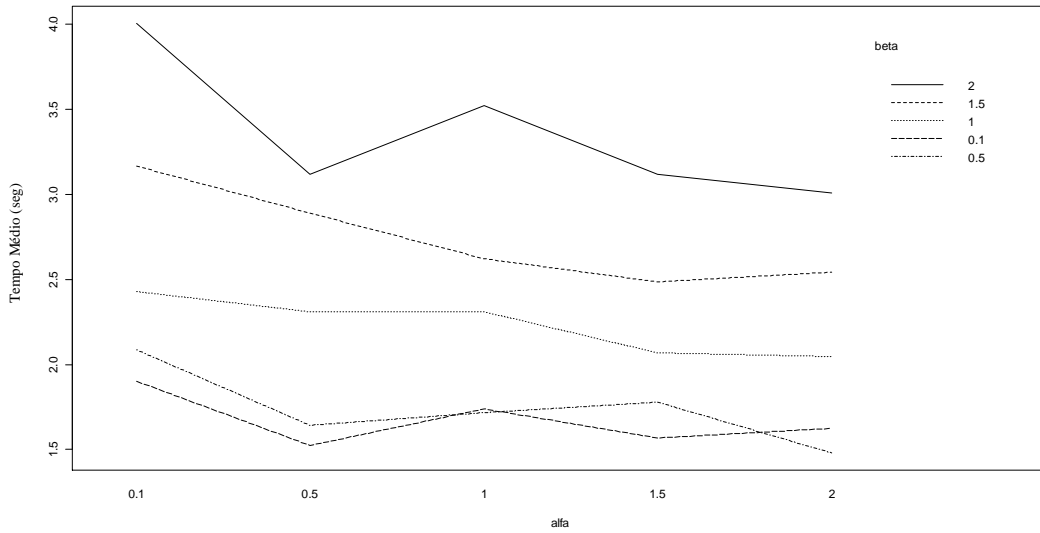


Figura 10 – Interação entre os parâmetros  $\alpha$  e  $\beta$ .

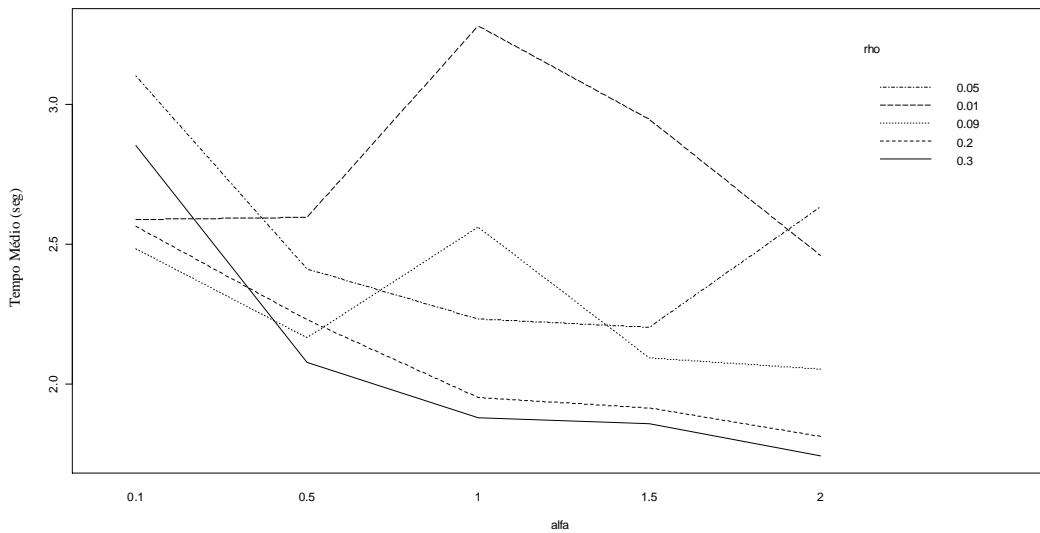


Figura 11 – Interação entre os parâmetros  $\alpha$  e  $\rho$ .



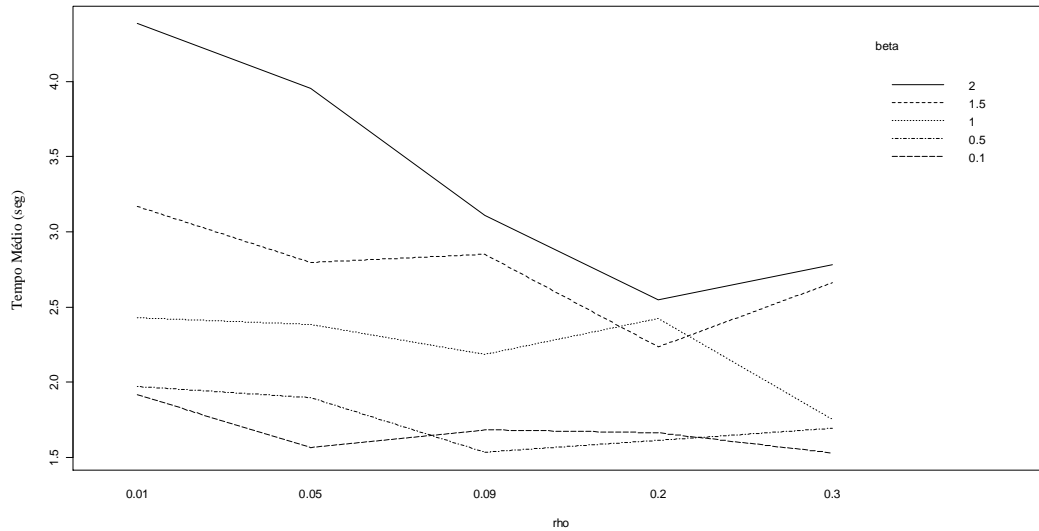


Figura 12 – Interação entre os parâmetros  $\rho$  e  $\beta$ .

Tabela 12 – Tempo médio de execução dos melhores valores paramétricos

| $\alpha$   | $\beta$    | $\rho$     | Tempo médio de execução (seg) |
|------------|------------|------------|-------------------------------|
| 0.5        | 0.1        | 0.09       | 1.925528                      |
| 1.5        | 0.1        | 0.05       | 1.679189                      |
| 1.5        | 0.5        | 0.2        | 1.562628                      |
| 2.0        | 0.1        | 0.2        | 1.581507                      |
| <b>2.0</b> | <b>0.1</b> | <b>0.3</b> | <b>1.173493</b>               |
| 2.0        | 0.5        | 0.09       | 1.429596                      |
| 2.0        | 0.5        | 0.3        | 1.799071                      |

Assim, ao final deste processo, tem-se que a melhor configuração paramétrica, com base na Tabela 12, foi  $\alpha = 2.0$ ,  $\beta = 0.1$  e  $\rho = 0.3$ .

## 9.4 Considerações Finais

Esta seção teve por objetivo demonstrar os passos para a configuração paramétrica para o algoritmo da ACO utilizada neste trabalho.

Devido ao grande número de parâmetros para a configuração, alguns valores paramétricos foram definidos empiricamente, para que fosse possível um ajuste mais rigoroso dos demais parâmetros.