# RESTABELECIMENTO DE SISTEMAS ELÉTRICOS DE POTÊNCIA COM A OTIMIZAÇÃO POR COLÔNIA DE FORMIGAS

**Tiago Silveira**
Universidade Federal de Alfenas (Unifal-MG)
Rua Gabriel Monteiro da Silva, 700, Alfenas - Brasil
tiago@bcc.unifal-mg.edu.br

**Luiz Eduardo da Silva**
Universidade Federal de Alfenas (Unifal-MG)
Rua Gabriel Monteiro da Silva, 700, Alfenas - Brasil
luizedu@bcc.unifal-mg.edu.br

**Germano Lambert-Torres**
Universidade Federal de Itajubá (Unifei)
Av. BPS, 1303, Itajubá - Brasil
germano@unifei.edu.bre-mail

**Humberto César Brandão de Oliveira**
Universidade Federal de Alfenas (Unifal-MG)
Rua Gabriel Monteiro da Silva, 700, Alfenas - Brasil
humberto@bcc.unifal-mg.edu.br

## RESUMO

Este trabalho apresenta uma alternativa evolutiva para a resolução do problema de Restabelecimento de Sistemas Elétricos de Potência, através da meta-heurística Otimização por Colônia de Formigas (ACO). A ACO foi utilizada para testar, após a ocorrência de uma falha, a ordem de religamento de chaves disjuntoras de um sistema elétrico, de modo que se fizesse o menor número de mudança de chaves para o seu restabelecimento, resultando, assim, em uma maior economia de recursos necessários para essa tarefa. A implementação da ACO seguiu uma das estratégias mais bem sucedidas desta meta-heurística, denominada *Max-Min Ant System* (*MMAS*), contando com algumas estruturas de dados adicionais, a fim de maximizar o seu desempenho. Ao final, são mostrados os resultados experimentais para o modelo idealizado, evidenciando a eficiência do modelo.

**PALAVRAS-CHAVE. Otimização por Colônia de Formigas. Restabelecimento de Sistemas Elétricos de Potência. Inteligência de Enxames. PO na Engenharia Elétrica.**

## ABSTRACT

This paper presents an evolutionary alternative to solving the problem of restoration of electric power systems, through the Ant Colony Optimization (ACO) meta-heuristic. The ACO was used to test, after the occurrence of a fault, the reclosing order of switches of an electrical system, in order to make the fewest of change of the keys to its restoration, thus resulting in a greater economy of resources to this task. The implementation of the ACO followed one of the most successful strategies of this meta-heuristic, called *Max-Min Ant System* (*MMAS*), using some additional data structures, in order to maximize its performance. Finally, the experimental results are shown for the idealized model, demonstrating the efficiency of the model.

**KEYWORDS. Ant Colony Optimization. Swarm Intelligence. Restoration of Electrical Power System. Operational Research in Electrical Engineering.**

# 1. Introdução

Atualmente, 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 confiáveis e disponíveis para a distribuição da energia produzida. 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 manipulação destes sistemas é cada vez mais comum, devido aos benefícios proporcionados, tais como rapidez, segurança e economia para sua operação. Entre os métodos para a manipulação destes sistemas, se tem como exemplos a Busca Tabu (Glover, F., 1989; Glover, F., 1990), o *Simulated Annealing* (Kirkpatrick *et al.*, 1983) e os algoritmos evolucionários, com as técnicas baseadas em inteligência de enxames, como a Otimização por Enxame de Partículas (Kennedy e Eberhart, 1995) e a Otimização por Colônia de Formigas ou ACO (do inglês, *Ant Colony Optimization*) (Dorigo, 1992). Esta última é uma técnica bastante interessante e será utilizada neste trabalho.

Por sua proposta original de trabalhar com problemas de caráter combinatório, a ACO é indicada para proceder com a manipulação de SEPs, 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 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, devido à quantidade exponencial de possibilidades. Logo, a adoção de uma heurística de otimização, como a ACO, é indispensável para se obter uma resolução factível em um tempo viável.

Assim, a ACO foi utilizada para testar a ordem de religamento de chaves de um sistema elétrico de modo que se fizesse o menor número possível de mudança de chaves para o restabelecimento de energia, resultando, assim, em uma maior economia de recursos necessários para essa tarefa. Para comprovar a eficácia do método proposto para a aplicação no problema descrito, foram realizados testes sobre uma instância de um SEP, denominada "*Distribution_System_01*" (Ramírez-Rosado e Bernal-Agustín, 1998), apresentando as possíveis soluções para vários casos de falhas.

O restante do trabalho está dividido da seguinte forma: a Seção 2 aborda as características teóricas da ACO; a Seção 3 trata mais formalmente do problema de restabelecimento de SEPs, apresentando suas peculiaridades; a Seção 4 apresenta a arquitetura do sistema, ou seja, o modo em que o problema de restabelecimento foi estruturado para o modelo de feromônio da ACO; a Seção 5 apresenta algumas estruturas de dados adicionais para esta tarefa, que auxiliaram a otimização pelo algoritmo idealizado; a Seção 6 mostra as configurações paramétricas e experimentais adotadas para a resolução do problema; a Seção 7 apresenta os resultados obtidos com o modelo proposto; por fim, a Seção 8 faz uma conclusão do estudo, apresentando, também, as futuras diretrizes do trabalho.

# 2. Otimização por Colônia de Formigas

A Otimização por Colônia de Formigas (ACO) é uma meta-heurística de inteligência de enxames inspirada no comportamento de agentes (formigas artificiais) em busca de alimento.

Originalmente descrita por Dorigo (1992), tem como idéia primordial a comunicação indireta entre seus indivíduos, através de uma trilha de feromônio artificial feita por cada formiga durante a exploração do espaço de busca. Dessa forma, a ACO mostra-se como um processo de aprendizado distribuído em que simples agentes, as formigas, trabalham de forma coletiva onde, adaptativamente, modificam a forma como o problema é representado e percebido por outras formigas (Dorigo e Stützle, 2002).

Devido ao seu promissor desempenho, a ACO se tornou um dos exemplos mais bem-sucedidos de sistemas de inteligência de enxames (Bonabeau, Dorigo e Théraulaz, 1999) e, com isso, tem sido aplicada em vários tipos de problemas, desde problemas de natureza estática (no qual a topologia e custo do problema não mudam durante a otimização), como o clássico Problema do Caixeiro Viajante, até 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, a qual, por sua proposta original de trabalhar com um conjunto de soluções discretas, envolve basicamente uma busca de caminho em grafos para a construção de soluções ao longo do processo de otimização.

Visando explicitar o seu funcionamento, em seguida serão apresentados os principais conceitos relacionados à implementação da meta-heurística ACO.

## 2.1. Idealização da Meta-Heurística da ACO

A maneira de modelar um problema de otimização combinatória, utilizando uma abordagem da ACO, é associá-lo a um modelo de feromônio, de forma que seja emulado o comportamento de formigas.

Diante disso, com base no trabalho de Deneuborg *et al.* (1990), 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 e Rinaldi, 1997). A primeira aplicação com um algoritmo de ACO foi feita usando este problema como teste (Dorigo, Caro e 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, no qual 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 (1):

$$p_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{l \in N_i^k}[\tau_{il}(t)]^{\alpha}[\eta_{il}]^{\beta}} \qquad (1)$$

na 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 todas as vizinhanças factíveis *N* da formiga *k*.

Além disso, a trilha de feromônio evapora por uma constante $\rho$, conforme equação (2), para evitar a convergência de todas as formigas para um caminho sub-ótimo.

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ij}(t) + \sum_{k=1}^{m}\Delta\tau_{ij}^{k}(t) \qquad (2)$$

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

Este processo de escolha de caminho, depósito e evaporação de feromônio é repetido até que todas as formigas tenham encontrado um caminho. O melhor caminho é escolhido e uma nova iteração acontece. O algoritmo termina depois de $n$ iterações, ou até que um tempo limite tenha sido alcançado. Ao final, o melhor caminho encontrado em todas as iterações é escolhido como a melhor solução encontrada pela colônia de formigas para o problema.

## 2.2. *Max-Min Ant System*

Uma das implementações mais bem sucedidas da meta-heurística ACO é o algoritmo denominado *Max-Min Ant System* (Stützle e 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 (3):

$$\tau_{ij}(t+1) = [(1-\rho).\tau_{ij}(t) + \Delta\tau_{ij}^{best}]_{\tau_{\min}}^{\tau_{\max}} \qquad (3)$$

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

# 3. Sistemas Elétricos de Potência

Sistemas Elétricos de Potência (SEPs) consistem em sistemas que englobam as unidades de geração, transmissão e distribuição da energia produzida (Zanetta Jr., 2005). Para atingir a meta proposta, este trabalho irá focar exclusivamente nos assuntos referentes à unidade de transmissão de um SEP, que 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.

Nos dias atuais, com o aumento da demanda por energia elétrica, torna-se fundamental também a demanda por SEPs que sejam confiáveis e disponíveis (Liu e Gu, 2006; Rossi, 2000). Porém, é impossível garantir que um SEP estará sempre disponível, devido à 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 desses sistemas sejam reparados ou substituídos, para que estes retornem a um estado de distribuição de energia similar ao estado 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 suas cargas, permitindo diversas configurações operacionais para a alimentação destas. 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. À tarefa de

reconfiguração deste sistema para atingir esse objetivo é dada o nome de Serviço de Restabelecimento (Watanabe, 2005).

Com isso, algumas características devem ser respeitadas em um SEP durante a atividade de correção de uma falha pelo serviço de restabelecimento. Tais características estão embutidas no problema de Restabelecimento de SEPs, e serão discutidas.

## 3.1. Restabelecimento de Sistemas Elétricos de Potência

O serviço de restabelecimento envolve a escolha de uma seqüência 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, após a ocorrência de uma falha. Para isso, os SEPs contam 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 uma linha do sistema.

Para o serviço de restabelecimento dos SEPs, devem ser observadas as seguintes características (Watanabe, 2005):

- *Restrição radial da rede*: devido à topologia dos SEPs ser similar a de uma árvore geradora de um grafo, o número de linhas que transferem energia, de modo que todas as cargas sejam atendidas, deve 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;
- *Restrição da capacidade da linha*: o fluxo de energia por uma linha não deve ser maior que sua capacidade.

Tais características são obrigatórias nos SEPs que apresentam 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 1. A Figura 1a ilustra um SEP em seu funcionamento normal, com algumas chaves normalmente abertas ($NO_i$), e as outras fechadas ($NC_i$), para que todas as cargas sejam supridas. A Figura 1b ilustra uma situação de falha. A Figura 1c 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 1d ilustra uma possível solução, caso a linha com a chave $NC_2$ da Figura 1c tenha uma sobrecarga.

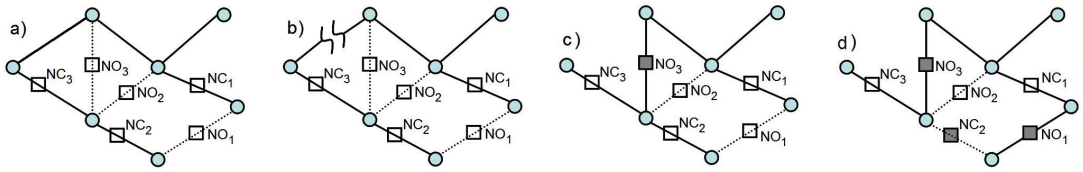



**Figura 1 – Representação de um Sistema Elétrico de Potência (a), assim como situações de restabelecimento (b,c,d) em caso de falhas no sistema.**

Um fato que pode ser observado na Figura 1 é o número de chaves alteradas no caso de uma falha. Devido à 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. Logo, 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 de chaves possíveis para formar uma sequência que satisfaça às restrições do problema. Diante desta natureza multi-objetiva, todo esse processo poderá ser dispendioso, o que torna o restabelecimento de SEPs uma tarefa de computação difícil. Este é o ponto que torna interessante a aplicação de heurísticas de otimização para problemas combinatórios difíceis.

# 4. Arquitetura do sistema

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

## 4.1. Formigas

As formigas da colônia serão o núcleo da técnica proposta, pois são as responsáveis pela construção de uma solução para o SEP em caso de ocorrência de uma falha.

Assim, irão adotar o seguinte processo de construção de soluções: na elaboração de uma solução, cada formiga vai possuir uma sequência de chaves $S = \{C_i, C_{i+1}, ..., C_n\}$. Entre estas chaves, estão embutidas os dois tipos de chaves já comentadas, as normalmente abertas (NO) e as normalmente fechadas (NC). Para que a característica radial da rede seja sempre satisfeita, é obrigatória a seguinte ordem para o restabelecimento: fechamento de uma chave NO e verificação da condição de parada; caso haja sobrecarga, abre-se uma chave NC, fecha-se uma NO e verifica-se novamente a condição de parada; esse processo segue até que se encontre uma solução. Diante disso, o conjunto $S$ pode ser ordenado da seguinte forma: para todo $i$ ímpar do conjunto, o elemento referenciado corresponderá a uma chave NO que poderá ser fechada; e para todo $i$ par do conjunto, o elemento referenciado corresponderá a uma chave NC, que poderá ser aberta para a construção de uma solução. 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.

## 4.2. Informação Heurística

Como as formigas detêm um processo de construção estocástico e devido à baixa influência das trilhas iniciais de feromônio, guiá-las para uma possível solução factível é algo indispensável no início da otimização.

Um ponto importante que se deve ter consciência para este problema é que não há nenhuma 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 Problema do Caixeiro Viajante, 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 devem ser fechadas e quais chaves NC 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 o sistema não será 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 do enxame durante a otimização.

## 4.3. Feromônio

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 $j$.

Assim, quanto mais feromônio possui uma chave do SEP, seja ela qual for (NO ou NC), tem-se a indicação de que várias formigas estão utilizando-a no restabelecimento 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 de restabelecimento.

# 5. Estruturas de dados adicionais

A intenção de utilizar um algoritmo de otimização em um problema real tem a finalidade de obter uma resposta rápida e interessante para o problema em questão. Além da utilização exclusiva de um algoritmo para auxiliar na resolução de um problema, pode-se recorrer a algumas estruturas de dados que modelem o problema a ser tratado. Isto torna mais natural e específica a representação deste para a estrutura que o algoritmo necessita para sua execução, melhorando ainda mais o seu desempenho.

Assim sendo, este trabalho buscou utilizar de estruturas de dados adicionais para representação do problema do SEP para o modelo de feromônio da ACO. Para isso, foram utilizadas duas estruturas de dados, que são: Árvore Geradora e Árvore Reversa.

## 5.1. Árvore Geradora

A forma que a Árvore Geradora foi concebida para o sistema, com base em Silva *et al.* (2010), pode ser observada na Figura 2.

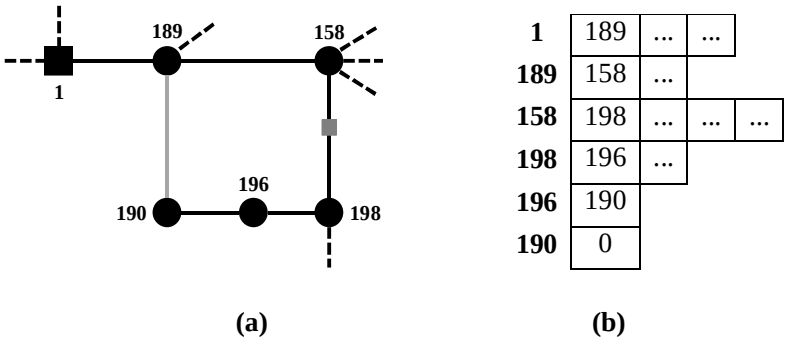


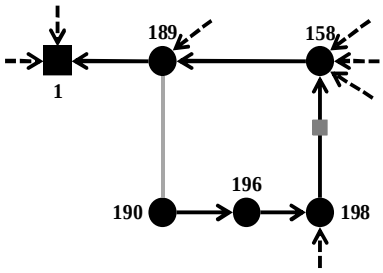

**Figura 2 – Representação da árvore geradora para o SEP, sendo (a) o próprio SEP e (b) a sua árvore geradora.**

Pela Figura 2, podem ser observadas as seguintes características: a Figura 2a representa uma instância de um SEP qualquer; a Figura 2b modela a Árvore Geradora para o SEP apresentado. Na estrutura da Árvore Geradora (2b), 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, são representadas as chaves filhas, que são as chaves que nas quais existem caminhos de conexão entre ambas. A carga que possui ligação para uma outra de número zero (0), indica que esta é uma carga terminal do sistema.

Dessa forma, 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 uma busca em largura na árvore. Logo, esta estrutura será elementar durante a construção de uma solução para uma formiga, indicando se as soluções elaboradas satisfazem à condição de Árvore Geradora.

## 5.2. Árvore Reversa

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 formal, se cada barra é identificada por um valor inteiro no intervalo [1..n], então v[$i$], $i \in \{1,...,n\}$ armazena o inteiro que identifica o pai de $i$ na Árvore Geradora. Para um melhor entendimento, a Figura 3 ilustra esta estrutura.




| 1 | 189 | 158 | 198 | 196 | 190 |
|---|---|---|---|---|---|
| 0 | 1 | 189 | 158 | 198 | 196 |

**(a)** **(b)**

**Figura 3 – Representação da árvore reversa para o SEP, no qual (a) mostra sua representação gráfica e (b) sua representação estrutural.**

Pela Figura 3, são observadas as seguintes características: a Figura 3a mostra, graficamente, a Árvore Reversa do SEP da Figura 2a; pela Figura 3b, é obtida esta mesma árvore, porém pela forma estrutural utilizada pelo algoritmo. Novamente, a 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, à fonte de geração.

Por fim, a utilização desta estrutura será importante na ocorrência de uma falha, pois permite ao algoritmo percorrer de forma eficiente as 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 um caminho alternativo pelas formigas do algoritmo.

# 6. Configurações do Sistema

Para experimentar o problema abordado neste trabalho, foi utilizada uma instância de um SEP 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 4.

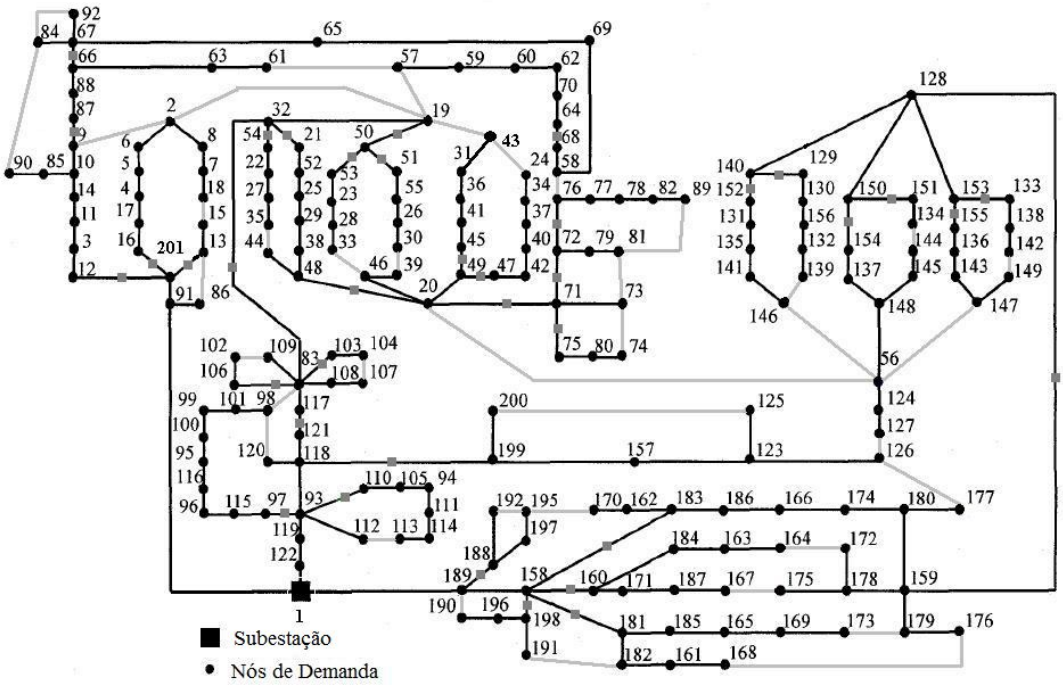



**Figura 4 – Instância *"Distribution_System_01"*.**

O algoritmo da ACO também terá suas configurações básicas que serão utilizadas durante sua execução. Logo, os valores dos parâmetros utilizados pela ACO estão identificados na Tabela 1.

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

| Parâmetro | Valor |
|---|---|
| $\alpha$ | *2* |
| $\beta$ | *0.1* |
| $\rho$ | *0.3* |
| $\tau_{max}$ | *0.999* |
| $\tau_{min}$ | *0.001* |
| Máx. Tentativas | *1* |
| Núm. Iterações | *50* |
| Núm. Formigas | *30* |

É importante observar que os três primeiros parâmetros ($\alpha$, $\beta$ e $\rho$) 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. Os demais parâmetros indicados foram definidos empiricamente para o experimento.

Para as configurações do *hardware* no qual foi executado o experimento, foi utilizado um computador com processador Core 2 Duo de 2GHz, com memória 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.

Adotando tais configurações e condições estabelecidas, esses parâmetros foram a base com que o algoritmo idealizado foi testado sobre o problema do SEP descrito.

# 7. Resultados e Discussões

Adotando as configurações da seção anterior, os resultados obtidos da execução do algoritmo proposto são apresentados, sendo também discutidos os efeitos desta implementação.

A Tabela II apresenta os resultados obtidos para várias situações de falhas no sistema, escolhidas aleatoriamente. A primeira coluna mostra uma situação de falha. A segunda coluna indica o porcentual de cargas atendidas com a situação de falha. As colunas 3 e 4 apresentam a melhor sequência de chaves normalmente aberta (NO) e normalmente fechada (NC) encontrada pelo algoritmo, respectivamente, sendo possível uma situação de falha possuir várias soluções. A coluna 5 indica o melhor tempo alcançado pelo algoritmo para encontrar a respectiva solução. Por fim, as 3 últimas colunas mostram alguns dados estatísticos relacionados à uma situação de falha que possui uma resolução com o mesmo número de alterações de chaves.

**Tabela 2 – Resultados das simulações de falhas no sistema**

| *Exemplos* | | *Solução Encontrada* | | | *Análise Estatística* | | |
|---|---|---|---|---|---|---|---|
| Linha com Falha | P% | NO | NC | t(s) | Tempo médio | Melhor tempo | Desvio Padrão |
| 1 – 122 | 54.51 | (2 – 19) (126 – 177) | 117 – 121 | 1.677 | 1.290 | 0.394 | 0.639 |
| | | (2 – 19) (126 – 127) | 117 – 121 | 0.884 | | | |
| | | (20 – 56) (2 – 19) | 20 – 48 | 1.270 | | | |
| | | (19 – 57) (126 – 127) | 117 – 121 | 0.394 | | | |
| | | (58 – 76) (126 – 127) | 117 - 121 | 0.396 | | | |
| | | (126 – 177) (19 – 57) | 117 – 121 | 2.428 | | | |
| | | (126 – 177) (58 – 76) | 117 – 121 | 0.789 | | | |
| 1 – 189 | 62.87 | (126 – 127) (58 – 76) | 117 – 121 | 0.389 | 1.373 | 0.389 | 0.526 |
| | | (126 – 177) (58 – 76) | 117 – 121 | 0.406 | | | |
| 1 – 91 | 82.62 | (2 – 19) (20 – 56) | 20 – 48 | 1.726 | 4.353 | 1.726 | 2.145 |
| | | (19 – 57) (20 – 56) | 20 – 48 | 2.898 | | | |
| | | (2 – 19) (20 – 56) (58 – 76) | (20 – 71) (21 – 32) | 3.023 | 4.531 | 2.612 | 2.698 |
| | | (19 – 57) (20 – 56) (58 – 76) | (20 – 71) (21 – 32) | 2.612 | | | |
| 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 | 0.001 |
| | | 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 |
| 128 – 140 | 93.62 | 56 – 146 | – | 0.046 | 0.047 | 0.046 | 0.001 |
| 10 – 85 | 99.10 | 84 – 90 | – | 0.046 | 0.046 | 0.046 | 0.001 |

Pela Tabela 2, pode ser observado que o algoritmo proposto sempre obteve uma

solução válida para uma respectiva falha, sendo, na maioria das vezes, em um intervalo médio de tempo menor do que 1 segundo. As únicas exceções estão a cerca das falhas “1 – 122”, “1 – 189” e “1 – 91”. Para esta última, 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 (5 alterações de chaves). Para o melhor tempo e o desvio padrão de restabelecimento das falhas, os resultados seguiram um comportamento semelhante ao tempo médio de restabelecimento.

Uma característica que deve ser observada concerne à heurística utilizada para escolha das chaves NO. Como a preferência é por aquelas chaves mais próximas à região de falha, as situações que necessitam de apenas uma alteração de chave para satisfazer o problema tiveram os melhores resultados. Isso ocorre, pois, para estes casos, 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.

Assim, como pode ser observado nos resultados, a meta-heurística ACO apresentou um desempenho satisfatório sobre a instância utilizada, visto a complexidade do problema tratado, sendo uma estratégia fortemente indicada para resolução de problemas desta natureza.

# 8. Conclusão e trabalhos futuros

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. A implementação da ACO 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, confirmando a ACO como uma promissora alternativa quando aplicada em problemas combinatórios da área da Engenharia Elétrica.

Como trabalhos futuros, a meta será testar o modelo implementado para outras instâncias da literatura de SEPs. A comparação com outras estratégias da meta-heurística ACO também será elaborada, a fim de verificar a influência de cada uma neste problema. Por fim, a adição da Lógica Paraconsistente ao modelo idealizado será feita para auxiliar as formigas na tomada de decisão, visando melhorar a eficiência do método.

# Agradecimentos

Os autores agradecem ao PROBIC da Universidade Federal de Alfenas e ao Grupo de Aplicação de Inteligência Artificial (GAIA) da Universidade Federal de Itajubá, pelo suporte no desenvolvimento deste trabalho.

# Referências

**Bonabeau, E., Dorigo, M. e Théraulaz, G.**, From Natual to Artificial Swarm Inteligence, *Oxford University Press*, 1999.

**Deneuborg, J. L., Aron, S., Goss, S. e Pasteels, J. M.** The self-organizing exploratory pattern of the argentine ant. *Journal of Insect Behaviour*, 3:159-168, 1990.

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

**Dorigo, M., Caro, G. e Gambardella, L. M.**, Ant Algorithms for Discrete Optimization, *Technical Report 98-10, IRIDIA*, Université Libre de Bruxelles. Submitted to Artificial Life, 1998.

**Dorigo, M. e 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. e 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. e 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. e Vecchi, M. P.**, Optimization by simulated annealing, *Science* 220 (4598) 671–680, 1983.

**Liu, Y. e 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. e 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, Itajubá, Brasil, 2000.

**Silva, L. E., Lambert-Torres, G., Martins, H. G., Coutinho, M. P., Silva, L. E. B. e Neto, J. C.,** An Application of ACO in System Reconfiguration. *2010 IEEE PES Transmission and Distribution*, 2010.

**Stützle, T. e Hoos, H. H.**, MAX-MIN Ant System, *Future Generation Computer Systems*. 16(8):889-914, 2000.

**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. *Livraria da Física*, São Paulo, 2005.