

# Algoritmo Genético aplicado ao problema de redução de perdas em redes primárias de distribuição

J.T. Belotti, Doutorando em Ciência da Computação com o RA 230260.

**Resumo.** A reconfiguração de chaves de uma rede de distribuição pode permitir uma redução das perdas técnicas dessa rede. Esse trabalho aplicou Algoritmos Genéticos para o problema da redução de perdas em redes primárias de distribuição. Foram propostas e aplicadas três estratégias de seleção dos pais: roleta, torneio e amostragem universal estocástica; dois tipos de cruzamento: de ponto e uniforme; dois tipos de mutação: estática e adaptativa; também foi aplicado o elitismo na seleção da nova população e adicionada uma busca local para refinar as soluções. Foram realizados experimentos computacionais com 8 instâncias de teste com redes variando de 13 a 10476 vértices. Os resultados obtidos mostraram uma redução na perda de energia de até 65%, comprovando que os Algoritmos Genéticos são uma excelente estratégia para resolver esse problema.

**Palavras-chave.** Redução de perdas por reconfiguração, Algoritmo Genético, Busca Local.

## 1. Introdução

Sistemas elétricos são divididos em três etapas: geração, transmissão e distribuição [1]. A geração é a responsável por transformar energia potencial em energia elétrica, no ano de 2017, 65,2% de toda energia elétrica produzida no Brasil foi resultado da geração hidráulica [2]. Após a energia ser gerada é necessário que ela seja transmitida dos centros de geração para os centros consumidores, essa tarefa fica a cargo da segunda etapa, a transmissão. Por fim, a distribuição é a etapa responsável por entregar diretamente a energia elétrica produzida e transmitida para os consumidores finais.

O transporte de energia elétrica, seja na etapa de transmissão ou de distribuição gera inevitavelmente perdas técnicas de energia, seja em função da transformação da energia elétrica em térmica que ocorre nos condutores, pelas perdas que ocorrem no núcleo dos transformadores ou ainda pelas perdas dielétricas [3]. Além disso, ainda existem as perdas não técnicas, também chamadas de perdas comerciais, elas ocorrem principalmente em virtude de fraudes e furtos por partes dos consumidores, além de erros de medição e de faturamento [3].

A Agência Nacional de Energia Elétrica (ANEEL) informou através da Superintendência de Gestão Tarifária (SGT) que no ano de 2016 cerca de 14% de toda

energia injetada na rede de distribuição brasileira foi perdida em forma de perdas técnicas e comerciais representando um prejuízo de 13,3 bilhões de reais [4]. Mais especificamente perdas técnicas ocorrem sobre 7,27% de toda energia elétrica inserida nas linhas de distribuição, com um prejuízo associado de 6,9 bilhões de reais apenas no ano de 2016 [4]. Este trabalho tratará especificamente de perdas técnicas na distribuição de energia elétrica.

Redes de distribuição são organizadas de forma radial, ou seja, existe apenas um caminho energizado entre a subestação de distribuição e cada um dos postos de consumo final, utilizando a linguagem matemática redes de distribuição são árvores, não possuindo ciclos. A Figura 1 apresenta um exemplo de uma rede de distribuição.

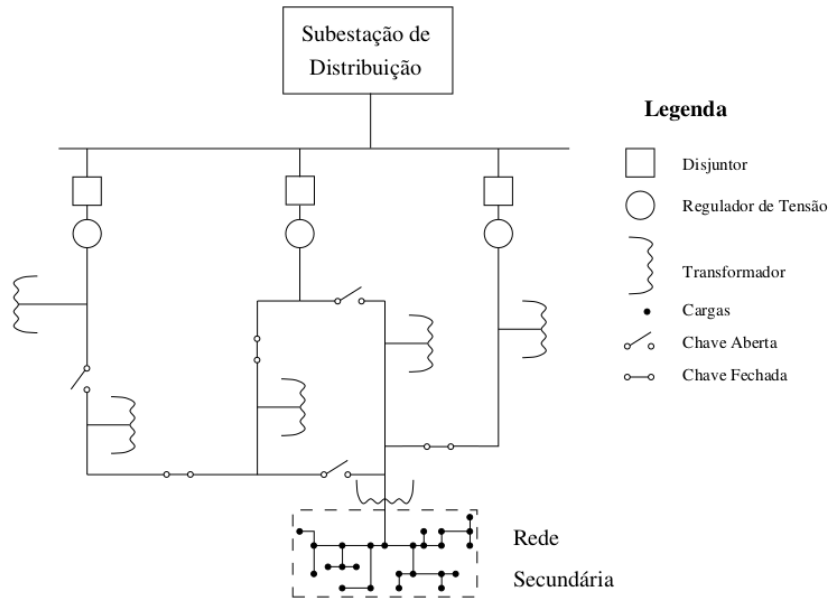


Figura 1: Exemplo de rede de distribuição (retirado de [5]).

Observando a Figura 1 nota-se a subestação de distribuição, responsável por receber a energia das linhas de transmissão e distribuir para os consumidores finais, a rede primária de distribuição contendo transformadores, chaves, disjuntores e reguladores de tensão e por fim a rede secundária, onde se encontram os consumidores finais. Ainda analisando a Figura 1 é possível comprovar que existe apenas um caminho entre a subestação e os consumidores finais. Além disso ao observar a rede de exemplo com mais cuidado nota-se que é possível alterar o caminho percorrido pela energia elétrica através da alteração de estado das chaves da rede.

Inicialmente as chaves de uma rede foram projetadas para situações de emergência, possibilitando que eventuais falhas da rede sejam isoladas e que cargas possam ser temporariamente transmitidas para outras redes. Entretanto, em 1975 Merlin

e Back observaram que a troca de estados das chaves de uma rede poderia proporcionar oportunidades para redução de perdas técnicas [6]. Cabe ressaltar ainda que a operação de chaveamento em uma rede elétrica é considerada uma operação arriscada, podendo levar a instabilidades na rede que geram desligamentos inesperados. Logo uma vez definido o estado das chaves o ideal é que os mesmos não sejam alterados.

O objetivo deste projeto é a aplicação de Algoritmos Genéticos ao problema de redução de perdas em redes primárias de distribuição através da reconfiguração de chaves.

### 1.1. Formulação matemática

Para o problema da redução de perdas técnicas os principais componentes da rede de distribuição são: chaves, blocos de carga, linhas e subestações. A Figura 2 apresenta um exemplo de rede de distribuição contendo apenas esses elementos.

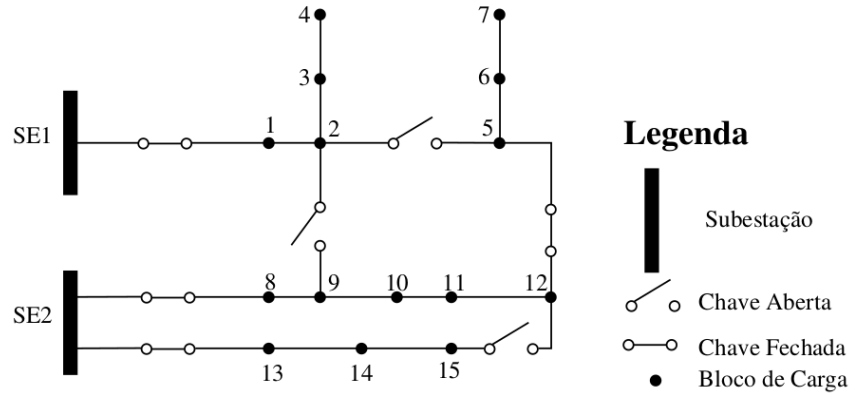


Figura 2: Rede de distribuição simplificada (retirado de [5]).

É possível realizar uma abstração da rede de distribuição através da sua representação por um grafo. Define-se um grafo  $G = (V, A)$  por um conjunto de vértices  $V(G)$  e um conjunto de arestas  $A(G)$ , sendo que uma aresta é um par não ordenado  $a = \{v_1, v_2\}$ ,  $v_1$  e  $v_2 \in V(G)$ . Nesse problema a direção que a energia se propaga na rede é importante, portanto as arestas serão representadas por pares ordenados  $(v_1, v_2)$ , onde uma aresta representa a direção de  $v_1$  para  $v_2$ . Um grafo  $G' = (V', A')$  é um subgrafo de  $G$  se  $V' \subseteq V(G)$  e  $A' \subseteq A(G)$ . A Figura 3 apresenta o grafo que representa a rede de distribuição simplificada da Figura 2. Note que foi incluído um nó R que é ligado diretamente às subestações de distribuição com a finalidade de facilitar o tratamento de aspectos de conectividade, além disso, as arestas que ligam o nó R às subestações tem impedância zero, ou seja, não influenciam nas perdas de energia. Mais detalhes sobre esse vértice R serão fornecidos na Seção 2.1.

As perdas em uma rede são calculadas com base nos fluxos de potência da rede, uma rede radial (sem ciclos) pode ter seus fluxos de potência calculados através das

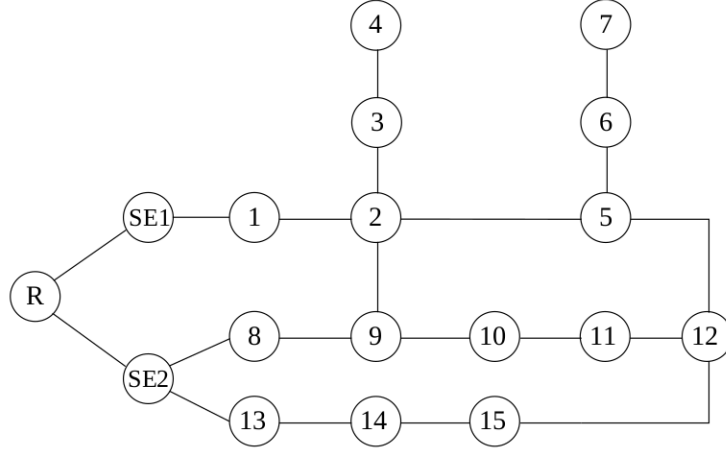


Figura 3: Grafo para a rede de distribuição da Figura2 (retirado de [5]).

equações de Baran e Wu [7]. As equações 1.1 e 1.2 apresentam simplificações das equações de Baran e Wu para o cálculo dos fluxos de potência. Vale mencionar que os fluxos devem ser calculados das folhas em direção às raízes da rede.

$$P_{v,s} = \sum_{a \in A_s} P_a + P_{Ls} \quad (1.1)$$

$$Q_{v,s} = \sum_{a \in A_s} Q_a + Q_{Ls} \quad (1.2)$$

tal que:

- $P_{v,s}$  é o fluxo de potência ativa entre os nós  $v$  e  $s$ ;
- $Q_{v,s}$  é o fluxo de potência reativa entre os nós  $v$  e  $s$ ;
- $P_{Ls}$  é a potência ativa da carga no nó  $s$ ;
- $Q_{Ls}$  é a potência reativa da carga no nó  $s$ ;
- $A_s$  é o conjunto de todas as arestas  $(s, v_i) \in A'(G')$ .
- $G' = (V, A')$  é uma árvore.

A partir do cálculo dos fluxos de rede pode-se definir o problema da redução de perdas em redes primárias de distribuição como o seguinte problema de otimização combinatória:

$$\begin{aligned}
& \text{Min} \quad \sum_{v \in V} \sum_{a \in A_v} r_a (P_a + Q_a)^2 \\
& \text{s.a.} \quad \begin{aligned}
& \underline{P} \leq P \leq \overline{P} \\
& \underline{Q} \leq Q \leq \overline{Q} \\
& \overline{G}' = (V, A') \text{ é uma árvore}
\end{aligned}
\end{aligned}$$

tal que  $r_a$  é a resistência da aresta  $a$ ,  $P_a$  e  $Q_a$  são, respectivamente, o fluxo de potência ativa e reativa da aresta  $a$ ;  $\overline{P}$ ,  $\underline{P}$ ,  $\overline{Q}$  e  $\underline{Q}$  indicam os limites superiores e inferiores para potência ativa e reativa nas arestas.

## 1.2. Revisão da literatura

Nara et al. [8] foram os primeiros na aplicação de Algoritmos Genéticos ao problema da redução de perdas de redes de distribuição, entretanto seu algoritmo sofria com indivíduos inactíveis. Posteriormente Lin et al. [9] e Ming et al. [10] conseguiram reduzir o problema de inactibilidade através de uma codificação baseada em malha. Entretanto o problema da inactibilidade só foi resolvido por Costa [11] através de uma nova representação que impossibilita a formação de indivíduos inactíveis. Tal representação é apresentada na Seção 2.1.

Em seu trabalho, Queiroz [5] fez uso de uma estratégia de busca local como parte de um algoritmo genético híbrido aplicado ao problema da redução de perdas. Os resultados obtidos mostraram que as estratégias de algoritmos genéticos alcançaram resultados muito próximos da melhor situação hipotética, dando ênfase ao GA com busca local.

Por sua vez, Cavalheiro [12] também aplicou algoritmos genéticos ao problema, dessa vez foi usada uma estratégia de evolução de chave aleatória tendenciosa (BRKGA). Os testes realizados em redes de benchmark mostraram que essa estratégia evolutiva gera bons resultados.

## 2. Metodologia

Algoritmos Genéticos podem ser definidos como algoritmos de busca estocástica baseados em mecanismos evolutivos, sendo um sistema computacional bio-inspirado, mais especificamente tendo como inspiração biológica a teoria da evolução [13]. Inicialmente é gerada uma população inicial em que cada indivíduo representa uma solução viável para o problema. Esses indivíduos são avaliados a fim de determinar qual o grau de adaptação de cada um, ou *fitness*. Em seguida são selecionados alguns indivíduos para serem combinados através do processo de cruzamento formando uma nova geração de indivíduos. Os indivíduos da nova geração sofrem uma mutação aleatória segundo uma determinada taxa e são então combinados com a geração anterior, algum critério de eliminação de indivíduos é aplicado a fim de se obter a nova população. Esses passos são repetidos até que um critério de pa-

rada seja alcançado [13]. O Algoritmo 1 apresenta todos os passos que descrevem o funcionamento do Algoritmo Genético.

---

<b>Algoritmo 1: ALGORITMO GENÉTICO.</b>	
<hr/>	
1	<b>início</b>
2	crie a população inicial
3	avale a população inicial
4	<b>repita</b>
5	selecione os pais
6	realize o cruzamento
7	<b>se</b> <i>condição de mutação for atendida</i> <b>então</b>
8	realize a mutação
9	reavale os indivíduos que sofreram mutação
10	<b>fim</b>
11	selecionar nova população
12	<b>até</b> <i>que a condição de parada seja atendida;</i>
13	<b>fim</b>

---

Como já foi mencionado na Seção 1, este projeto aplicou algoritmos genéticos ao problema de redução de perdas em redes primárias de distribuição. A metaheurística de algoritmo genético foi escolhida devido a sua robustez quando aplicada a problemas combinatórios, tendo apresentado bons resultados na literatura para problemas de difícil complexidade computacional, além da sua estrutura permitir que informações específicas sobre o problema possam ser utilizadas para melhorar seu desempenho.

A seguir, as seções de 2.1 até 2.7 detalham como cada aspecto do algoritmo genético foi modelado ao problema em questão.

## 2.1. Representação das soluções

Algoritmos genéticos representam os indivíduos através de uma cadeia de caracteres chamada de cromossomo, onde cada segmento do cromossomo, denominado gene, representa uma parte da solução. O problema estudado neste trabalho apresenta como peculiaridade o fato das soluções deverem ser árvores, de modo que apenas marcar com 0 ou 1 os nós que fazem parte da árvore torna as tarefas de cruzamento e mutação complexas, uma vez que será necessário verificar se não foram formados ciclos e corrigi-los caso necessário.

Para evitar esse tipo de problema este trabalho fez uso da representação NKR (do inglês *Network Random Keys*) criada por Costa [11] para garantir que cada cromossomo represente uma árvore e que não seja necessário qualquer tipo de cuidado extra no cruzamento ou na mutação. Além disso, também fez-se uso da representação CV (do inglês *Characteristic Vectors*) para facilitar a identificação se uma aresta faz parte da solução ou não. A Tabela 1 apresenta as duas codificações para o mesmo cromossomo.

Tabela 1: Cromossomo codificado por CV e NRK.

Arestas										
	1	2	3	4	5	6	7	8	9	10
CV	1	0	1	1	0	1	0	0	1	1
NRK	29	85	9	15	63	5	44	77	34	72

Na Tabela 1 note que cada gene do cromossomo representa uma aresta do grafo  $G = (V, A)$  e além disso tem um peso (NRK) e um valor binário (CV) associados. A árvore representada por esse cromossomo é a árvore geradora de custo mínimo obtida através dos pesos contidos nos genes do cromossomo. Para se obter essa árvore geradora de custo mínimo basta executar algum algoritmo de grafos como Prim ou Kruskal. A partir dessa árvore os valores de CV são determinados, toda aresta do grafo  $G$  que faz parte da árvore recebe o valor binário 1, caso contrário recebe 0. A Figura 4 apresenta a árvore geradora de custo mínimo extraída do cromossomo da Tabela 1.

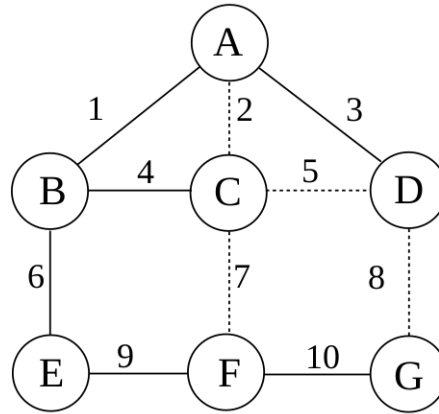


Figura 4: Árvore geradora de custo mínimo do cromossomo da Tabela 1.

Na Seção 1 foi mencionado que ao transformar a rede em um grafo deve-se adicionar um vértice R e ligá-lo a todas as subestações. Fazendo isso e inserindo pesos negativos nessas arestas garante-se que apenas uma árvore será extraída do cromossomo e a mesma conterá todas as subestações da rede.

Para a extração da árvore geradora de custo mínimo dos cromossomos foi implementado o algoritmo de Kruskal com Union-find para alcançar uma complexidade de  $O(A \log V)$  no pior caso, uma vez que se trata de grafos conexos.

A população inicial foi gerada através de uma heurística construtiva. Os pesos de cada indivíduo foram definidos por um vetor de números aleatórios no intervalo de  $[0, |A|]$  sem repetição. Gerando assim variedade para a população inicial.

## 2.2. Função de avaliação

A função de avaliação tem o objetivo de fornecer uma medida de qualidade para os indivíduos da população. Dado um cromossomo, a função de avaliação retorna um valor que é a medida de adaptação desse indivíduo em relação ao ambiente. Essa medida também é conhecida como *fitness*.

No problema de redução de perdas a função de avaliação dos indivíduos é a própria função objetivo do problema combinatório. Como se trata de um problema de minimização quanto menor for o valor do *fitness* de um indivíduo mais bem adaptado ao meio ele está.

## 2.3. Seleção dos pais

A seleção tem papel fundamental dentro dos algoritmos genéticos, pois é ela quem seleciona os indivíduos que serão combinados para formar a nova geração. Este trabalho fez a utilização de três estratégias de seleção: roleta, torneio e amostragem universal.

Na seleção por roleta cada indivíduo tem a probabilidade de ser selecionado proporcional ao seu valor de *fitness*, quando a roleta gira indivíduos com maior aptidão tem maiores chances de serem selecionados. Como o problema em questão é um problema de minimização, os menores valores de *fitness* são os com mais chances de serem selecionados, para facilitar a implementação propôs-se a utilização de um valor de *fitness* auxiliar  $f' = \frac{1}{1+fitness}$  na roleta.

A seleção por torneio elenca  $n$  indivíduos de forma aleatória e compara seus valores de *fitness*, é escolhido o indivíduo com o melhor *fitness*.

Por sua vez, a seleção por Amostragem Universal Estocástica é considerada a melhor maneira de selecionar os pais. Seu funcionamento é semelhante ao da roleta, entretanto agora existem  $n$  setas igualmente espaçadas que irão selecionar todos os pais de uma só vez.

## 2.4. Cruzamento

O cruzamento ou *crossover* é o responsável por combinar dois indivíduos para a formação de novos. Neste trabalho foram utilizados duas estratégias de cruzamento: cruzamento de ponto e cruzamento uniforme. Lembrando que em função da escolha da representação NRK não é necessário verificar se essa operação gerou algum ciclo.

O cruzamento de ponto é o mais simples, nele são escolhidos  $p$  pontos de ruptura, de modo que os filhos gerados recebem os materiais genéticos dos pais de forma alternada. Nesse trabalho optou-se pelo cruzamento de 2 pontos.

Por sua vez o cruzamento uniforme combina o material genéticos dos pais nos filhos de forma aleatória, cada gene dos filhos tem 50% de chance de vir de cada um dos pais, sendo escolhido de forma aleatória.



## 2.5. Mutação

A função do operador de mutação é introduzir perturbações nos indivíduos com o objetivo de manter a diversidade genética da população, proporcionando uma maior exploração do espaço de busca. Normalmente a mutação é aplicada a um gene do cromossomo, trocando seu valor por outro gerado de forma aleatória. A probabilidade de um gene de um cromossomo sofrer mutação é chamada de taxa de mutação, esse valor deve ser pequeno para que a solução não se degenere.

Nesse trabalho foi considerado como mutação a troca de estado de uma aresta, assim após a determinação de qual gene sofrerá a mutação é verificado na codificação CV se o mesmo faz parte da árvore ou não. Caso ele faça parte da árvore seu peso é alterado para o valor do maior peso, fazendo assim com que ele deixe de fazer parte da árvore. Caso contrário, se a aresta representada por esse gene não fizer parte da árvore seu peso deve ser alterado para 0, garantindo então que a mesma passará a fazer parte da solução.

Este trabalho aplicou duas estratégias de mutação: mutação estática e mutação adaptativa. A mutação estática utiliza a mesma taxa de mutação durante toda a execução do algoritmo, para esse problema foram utilizadas taxas de mutação de 10% e 20%. Por sua vez, a mutação adaptativa altera o valor da taxa de mutação dependendo do estado da população, quando a população possui pouca diversidade, a taxa é elevada e quando a diversidade é alta a taxa de mutação deve ser baixa. Nesse trabalho será utilizada a Equação 2.1 para atualizar a taxa de mutação a cada geração do algoritmo.

$$\text{tax} = 0,5 \cdot \left( 1 - \frac{\sigma}{\max_{c \in P} f(c)} \right) \quad (2.1)$$

tal que  $\sigma$  é o desvio padrão do fitness,  $c$  é um cromossomo da população  $P$  e  $f(c)$  é o fitness do cromossomo  $c$ .

## 2.6. Seleção da nova população

Normalmente algoritmos genéticos substituem a população pela nova população formada a cada geração do algoritmo. Uma forma de complementar essa seleção é através do elitismo, nele os  $n$  melhores indivíduos da geração anterior são mantidos e inseridos na nova população.

Nesse trabalho foram aplicadas três estratégias de seleção da nova população, a primeira realizará a substituição da população atual pela nova aplicando o elitismo de 1 elemento. A segunda junta os elementos recém criados com a população atual e seleciona os melhores para a próxima geração. Por fim, a terceira substitui totalmente a população atual pela nova.

## 2.7. Busca Local

Com o objetivo de aumentar a eficiência do algoritmo foi adicionada uma etapa de busca local ao GA. Essa operação faz buscas na vizinhança de uma solução,

procurando por outra configuração que resulte em uma menor perda. Para tal foram consideradas operações de troca entre uma aresta que faz parte da árvore de solução e outra que não faz, para realizar a troca basta inverter os pesos das arestas.

Entretanto vale mencionar que para verificar se um vizinho é melhor que a solução atual é necessário avaliá-lo, e no problema em questão a avaliação de uma solução não é uma tarefa barata, uma vez que é necessário extrair a árvore geradora de custo mínimo antes de calcular a perda da rede. Portanto, foi definido que a busca local será executada apenas a cada 100 gerações do algoritmo.

### 3. Experimentos computacionais

Para a execução dos experimentos computacionais foram utilizadas oito instâncias de teste, cada uma contendo uma quantidade de vértices, arestas e subestações. As instâncias de teste também fornecem a resistência de cada aresta e a carga de cada nó da rede. Todas as instâncias de teste estão disponíveis em [www.dejazzer.com/reds.html](http://www.dejazzer.com/reds.html) e são apresentadas na Tabela 2, nela é possível verificar o nome das instâncias, a quantidade de vértices, arestas e subestações, além do tamanho da população do GA que foi utilizado em cada instância.

Tabela 2: Instâncias de teste.

Intância	Vértices	Arestas	Subestações	Tam. população
bus_13_3	13	16	3	30
bus_29_1	29	30	1	30
bus_32_1	32	37	1	30
bus_83_11	83	96	11	30
bus_135_8	135	156	8	30
bus_201_3	201	216	3	30
bus_873_7	873	900	7	50
bus_10476_84	10476	10736	84	100

Todos os códigos desenvolvidos por esse trabalho foram implementados na linguagem de programação Java e encontram-se disponíveis em [https://github.com/jonatastbelotti/M0824A\\_projeto](https://github.com/jonatastbelotti/M0824A_projeto).

Ao todo foram testadas 10 configurações diferentes de Algoritmo Genético. Foi definida uma configuração padrão e cada configuração restante representa a variação de uma etapa do GA (seleção, cruzamento, mutação, etc.) para que seja possível comparar o desempenho de diferentes técnicas para cada etapa do algoritmo. As configurações testadas foram:

- **Conf1 (PADRÃO):** Seleção por Torneio, cruzamento de Ponto, mutação Estática com taxa de 10%, nova população substitui a antiga e sem Busca Local;
- **Conf2:** Seleção por Roleta;

- **Conf3:** Seleção por Amostragem Universal Estocástica;
- **Conf4:** Cruzamento Uniforme;
- **Conf5:** Mutação Estática com taxa de 20%;
- **Conf6:** Mutação Adaptativa;
- **Conf7:** Elitismo de 1 indivíduo;
- **Conf8:** Nova população se junta com a anterior e são selecionados os melhores;
- **Conf9:** Com Busca Local;
- **Conf10:** Seleção por Amostragem Universal Estocástica, cruzamento uniforme, mutação adaptativa, nova população se junta com a anterior e são selecionados os melhores e sem Busca Local;

A configuração 10 (Conf10) não se trata da variação de uma etapa do GA em relação a configuração padrão, mas sim de uma configuração composta pelas técnicas consideradas mais promissoras para cada etapa do algoritmo.

Foi utilizado como critério de parada para todos os experimentos um tempo de execução de 30 minutos e um limite de 1 milhão de gerações sem melhora. Todos os experimentos computacionais foram realizados em uma máquina com 8GB de memória RAM DDR3 1333MHz, processador Intel Core i5-2450M 2.5GHz x 4 e com sistema operacional Ubuntu 18.04 de 64-bits.

### 3.1. Resultados

A Tabela 3 apresenta os resultados obtidos por todas as configurações de GA testadas para a instância bus\_13\_3. Nela é possível verificar o valor da perda de potência original na rede sem qualquer modificação e o valor final alcançado pelos algoritmos, além do percentual de melhora de cada configuração.

Observando os dados da Tabela 3 verifica-se que todas as configurações alcançaram o mesmo resultado. Esse fato não causa espanto uma vez que essa é a menor instância do problema, contendo apenas 13 vértices. Mesmo se tratando da menor instância testada note que os GA conseguiram reduzir a perda em 6,9477%.

Por sua vez, a Tabela 4 apresenta os resultados obtidos por todas as configurações para a instância bus\_29\_1. Note que aqui mais uma vez não houve uma configuração que se destaca-se, novamente todas as 10 configurações testadas alcançaram o mesmo resultado, uma diminuição na perda de 0,0031%. Mais uma vez esse fato pode ser explicado pelo tamanho da instância, contendo apenas 29 vértices ainda se trata de uma instância pequena.

A Tabela 5 apresenta os resultados obtidos para a instância bus\_32\_1. Aqui pela primeira vez tivemos configurações que se destacaram das demais, destacadas em negrito temos as configurações 8, 9 e 10 que foram as que obtiveram as menores perdas para essa instância.

Tabela 3: Resultados para instância bus\_13\_3.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	113,9088	105,99475	6,9477%
<b>Conf2</b>	113,9088	105,99475	6,9477%
<b>Conf3</b>	113,9088	105,99475	6,9477%
<b>Conf4</b>	113,9088	105,99475	6,9477%
<b>Conf5</b>	113,9088	105,99475	6,9477%
<b>Conf6</b>	113,9088	105,99475	6,9477%
<b>Conf7</b>	113,9088	105,99475	6,9477%
<b>Conf8</b>	113,9088	105,99475	6,9477%
<b>Conf9</b>	113,9088	105,99475	6,9477%
<b>Conf10</b>	113,9088	105,99475	6,9477%

Tabela 4: Resultados para instância bus\_29\_1.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	0,013709432	0,013709	0,0031%
<b>Conf2</b>	0,013709432	0,013709	0,0031%
<b>Conf3</b>	0,013709432	0,013709	0,0031%
<b>Conf4</b>	0,013709432	0,013709	0,0031%
<b>Conf5</b>	0,013709432	0,013709	0,0031%
<b>Conf6</b>	0,013709432	0,013709	0,0031%
<b>Conf7</b>	0,013709432	0,013709	0,0031%
<b>Conf8</b>	0,013709432	0,013709	0,0031%
<b>Conf9</b>	0,013709432	0,013709	0,0031%
<b>Conf10</b>	0,013709432	0,013709	0,0031%

Tabela 5: Resultados para instância bus\_32\_1.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	55631859,18	40437254,585	27,3127%
<b>Conf2</b>	55631859,18	40569392,2075	27,0752%
<b>Conf3</b>	55631859,18	39954096,9075	28,1812%
<b>Conf4</b>	55631859,18	39221067,155	29,4989%
<b>Conf5</b>	55631859,18	38654432,925	30,5174%
<b>Conf6</b>	55631859,18	39954096,9075	28,1812%
<b>Conf7</b>	55631859,18	38654432,925	30,5174%
<b>Conf8</b>	<b>55631859,18</b>	<b>38558920,065</b>	<b>30,6891%</b>
<b>Conf9</b>	<b>55631859,18</b>	<b>38558920,065</b>	<b>30,6891%</b>
<b>Conf10</b>	<b>55631859,18</b>	<b>38558920,065</b>	<b>30,6891%</b>

A Tabela 6 contém os resultados dos testes computacionais para a instância

bus\_83\_11. Observando a tabela nota-se que a configuração com a menor perda foi a com uso da Busca Local (Conf9), tendo alcançado uma diminuição de 10,3596% na perda de energia da rede. Cabe ressaltar ainda que a configuração 8 teve uma piora no resultado, tendo um aumento de 4,5802% na perda de energia da rede da instância bus\_83\_11.

Tabela 6: Resultados para instância bus\_83\_11.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	123854986,7	117574888,85	5,0705%
<b>Conf2</b>	123854986,7	122905991,0	0,7662%
<b>Conf3</b>	123854986,7	118650549,11	4,2020%
<b>Conf4</b>	123854986,7	113073479,59	8,7049%
<b>Conf5</b>	123854986,7	119462784,21	3,5462%
<b>Conf6</b>	123854986,7	118893877,53	4,0055%
<b>Conf7</b>	123854986,7	122128604,39	1,3938%
<b>Conf8</b>	123854986,7	129527901,04	-4,5802%
<b>Conf9</b>	<b>123854986,7</b>	<b>111024024,11</b>	<b>10,3596%</b>
<b>Conf10</b>	123854986,7	112168575,68	9,4355%

Na Tabela 7 são apresentados os resultados obtidos para os experimentos computacionais da instância bus\_135\_8. Note que para essa instância a configuração com a menor perda de energia foi aquela que realizou a seleção dos pais por meio da Roleta, tendo uma diminuição na perda de energia da rede de 8,8760%. Esse resultado é surpreendente uma vez que a roleta é um mecanismo de seleção de pais extremamente tendencioso, tendendo a não manter a diversidade da população.

Tabela 7: Resultados para instância bus\_135\_8.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	97708466,33	93501699,539268	4,3054%
<b>Conf2</b>	<b>97708466,33</b>	<b>89035792,638231</b>	<b>8,8760%</b>
<b>Conf3</b>	97708466,33	89136798,323128	8,7726%
<b>Conf4</b>	97708466,33	96791747,7865	0,9382%
<b>Conf5</b>	97708466,33	96818755,473554	0,9105%
<b>Conf6</b>	97708466,33	97244928,233904	0,4744%
<b>Conf7</b>	97708466,33	92371092,785871	5,4625%
<b>Conf8</b>	97708466,33	123391280,82535	-26,2851%
<b>Conf9</b>	97708466,33	93569595,835209	4,2359%
<b>Conf10</b>	97708466,33	94979944,223239	2,7925%

A Tabela 8 contém os resultados obtidos pelos experimentos computacionais para a instância bus\_201\_3. Aqui tivemos mais uma vez a configuração 10 como destaque, tendo alcançado uma diminuição de 3,9249% na perda de energia da rede. Aqui ainda cabe ressaltar que várias configurações apresentaram uma piora

no resultado, com destaque especial para a configuração padrão, que resultou em um aumento de 76,3015% na perda de energia da rede. A melhora pouco expressiva da melhor configuração e a piora nos resultados das demais de maneira geral pode indicar que a rede da instância bus\_201\_3 apresenta mais dificuldade para ser reconfigurada do que as demais instâncias testadas até o momento, tal dificuldade pode se dar pelo fato da rede já estar bem configurada, gerando assim poucas opções de melhora.

Tabela 8: Resultados para instância bus\_201\_3.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	184509887,6	325293840,87621	-76,3015%
<b>Conf2</b>	184509887,6	184509887,639038	-2,1157e-08%
<b>Conf3</b>	184509887,6	183489356,761331	0,5531%
<b>Conf4</b>	184509887,6	267251964,477108	-44,8442%
<b>Conf5</b>	184509887,6	191552605,182492	-3,8169%
<b>Conf6</b>	184509887,6	180254401,29452	2,3063%
<b>Conf7</b>	184509887,6	226132879,540237	-22,5586%
<b>Conf8</b>	184509887,6	235414668,466637	-27,5891%
<b>Conf9</b>	184509887,6	180539954,324424	2,1516%
<b>Conf10</b>	<b>184509887,6</b>	<b>177267903,602767</b>	<b>3,9249%</b>

Por sua vez, a Tabela 9 apresenta os resultados obtidos para a instância bus\_873\_7. Diferentemente da instância anterior, onde as melhoras alcançadas pelos algoritmos foram inexpressivas ou negativas, agora todos as configurações testadas conseguiram reduzir a perda de energia da rede em pelo menos 47%, com especial destaque para a configuração 6, que faz uso da mutação adaptativa, tendo alcançado uma redução de 65,0718% na perda de energia da rede. Esses fatos levam a acreditar que se trata de uma rede pouco otimizada, portanto com muitas oportunidades de melhoria.

Tabela 9: Resultados para instância bus\_873\_7.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	45803330890	20576500979,213936	55,0764%
<b>Conf2</b>	45803330890	18039863508,737392	60,6145%
<b>Conf3</b>	45803330890	16699630231,703117	63,5405%
<b>Conf4</b>	45803330890	16600256473,504797	63,7575%
<b>Conf5</b>	45803330890	18359267826,918983	59,9171%
<b>Conf6</b>	<b>45803330890</b>	<b>15998265272,958487</b>	<b>65,0718%</b>
<b>Conf7</b>	45803330890	23888383471,162514	47,8457%
<b>Conf8</b>	45803330890	21409563021,696877	53,2576%
<b>Conf9</b>	45803330890	16809903946,876883	63,2998%
<b>Conf10</b>	45803330890	18934281958,67709	58,6617%

A Tabela 10 apresenta os resultados obtidos para a última instância testada, a instância bus\_1476\_84. Note que a linha dedicada a configuração 9 não contém resultados, isso se deve ao fato de que essa configuração não foi capaz de executar ao menos 1 geração completa do GA em 30 minutos. Se trata da configuração com Busca Local, e como já foi mencionado na Seção 2.7 essa é uma operação cara. A combinação dessa operação cara com a maior instância testada fez com que apenas 1 geração do GA levasse mais de 30 minutos para ser executada.

Tabela 10: Resultados para instância bus\_1476\_84.

GA	Perda Original	Fitness	% Melhora
<b>Conf1</b>	534527141823,62	601126855410,8429	-12,4595%
<b>Conf2</b>	534527141823,62	778714733253,3881	-45,6829%
<b>Conf3</b>	534527141823,62	495213551200,69867	7,3548%
<b>Conf4</b>	<b>534527141823,62</b>	<b>318968752519,4673</b>	<b>40,3269%</b>
<b>Conf5</b>	534527141823,62	434462299906,562	18,7202%
<b>Conf6</b>	534527141823,62	424325723812,9178	20,6166%
<b>Conf7</b>	534527141823,62	512008994975,2709	4,2127%
<b>Conf8</b>	534527141823,62	502933564227,95337	5,9105%
<b>Conf9</b>	534527141823,62	-	-
<b>Conf10</b>	534527141823,62	339252807064,8275	36,5321%

Ainda observando os dados da Tabela 10 verifica-se que a configuração com melhor desempenho foi a Conf4, a com uso do Cruzamento Uniforme, tendo reduzido a perda de energia da rede em 40,3269%.

Com o objetivo de facilitar as comparações de desempenho entre cada configuração de GA testada a Tabela 11 apresenta a quantidade de vez em que cada configuração obteve o melhor resultado.

Tabela 11: Resultados gerais.

GA	Descrição	Num. vitórias
<b>Conf1</b>	Padrão	2
<b>Conf2</b>	Seleção por Roleta	3
<b>Conf3</b>	Seleção por Amostragem	2
<b>Conf4</b>	Cruzamento Uniforme	3
<b>Conf5</b>	Mutação 20%	2
<b>Conf6</b>	Mutação Adaptativa	3
<b>Conf7</b>	Elitismo	2
<b>Conf8</b>	Seleciona melhores indivíduos	3
<b>Conf9</b>	<b>Busca Local</b>	<b>4</b>
<b>Conf10</b>	<b>Misto</b>	<b>4</b>

Do ponto de vista da seleção dos pais a Tabela 11 apresenta uma surpresa, a roleta foi a técnica com o maior número de melhores resultados. Já no que

diz respeito ao cruzamento a técnica que se demonstrou mais apropriada para o problema da redução de perdas foi o cruzamento uniforme. A mutação adaptativa foi melhor que as duas taxas testadas para a mutação estática. No que diz respeito a seleção da nova população a estratégia com melhores resultados foi juntar as duas populações e selecionar os melhores indivíduos.

Destacaram-se duas configurações dentre as 10 testadas, Conf9 e Conf10, as duas foram as melhores em 4 instâncias. Já era esperado esse resultado da configuração 10, uma vez que ela foi composta por todas as estratégias consideradas as melhores em cada parte do GA. A surpresa fica por conta da configuração 9, apenas adicionar uma etapa de busca local na configuração padrão foi capaz de ter o mesmo resultado que a configuração favorita. Esse resultado mostra que a intensificação de uma solução já existe por meio da aplicação de uma busca local gera ótimos resultados para o problema em questão.

Para colaborar com tudo que já foi mencionado a Figura 3 apresenta o gráfico Performance Profile para todos os experimentos realizados.

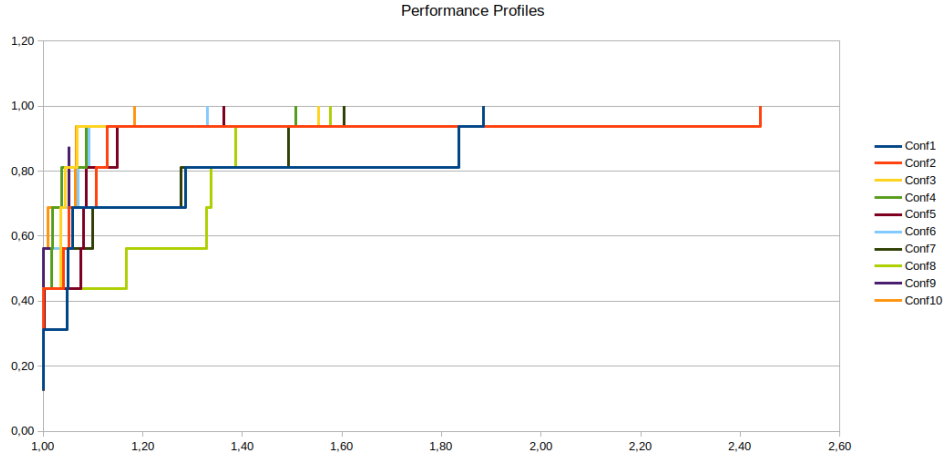


Figura 5: Gráfico Performance Profile.

Mais uma vez é evidenciado os bons resultados alcançados pelas configurações 9 e 10. Além disso, pelo gráfico da Figura 3 é fácil ver o quanto os resultados da roleta e da configuração padrão se distanciaram dos melhores resultados em algumas instâncias de teste.

Com o objetivo de proporcionar a visualização da reconfiguração da rede que resultou na redução da perda de energia a Figura 6 apresenta a rede original da instância bus\_32\_1 e a Figura 7 apresenta a rede da instância bus\_32\_1 após a reconfiguração proposta pela configuração 10 do GA que resultou em uma diminuição de 30,6891% da perda de energia.



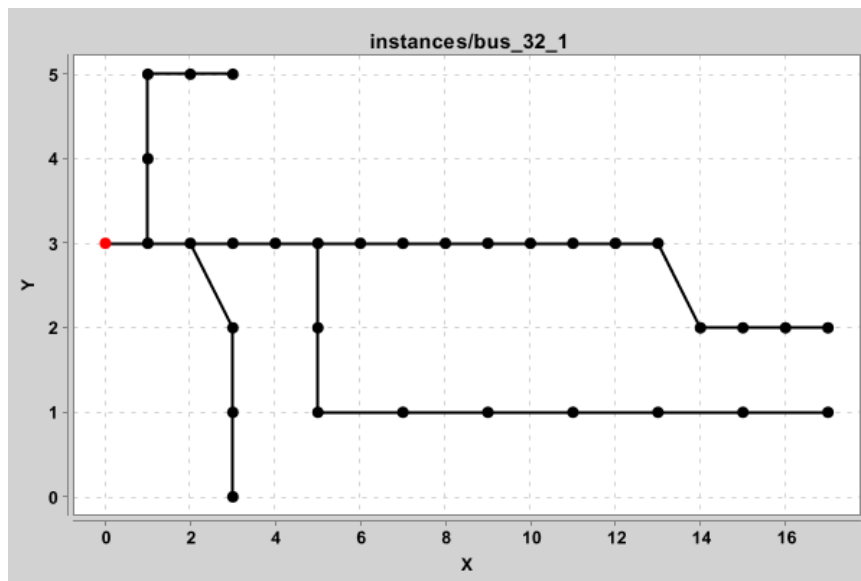


Figura 6: Rede original bus\_32\_1.

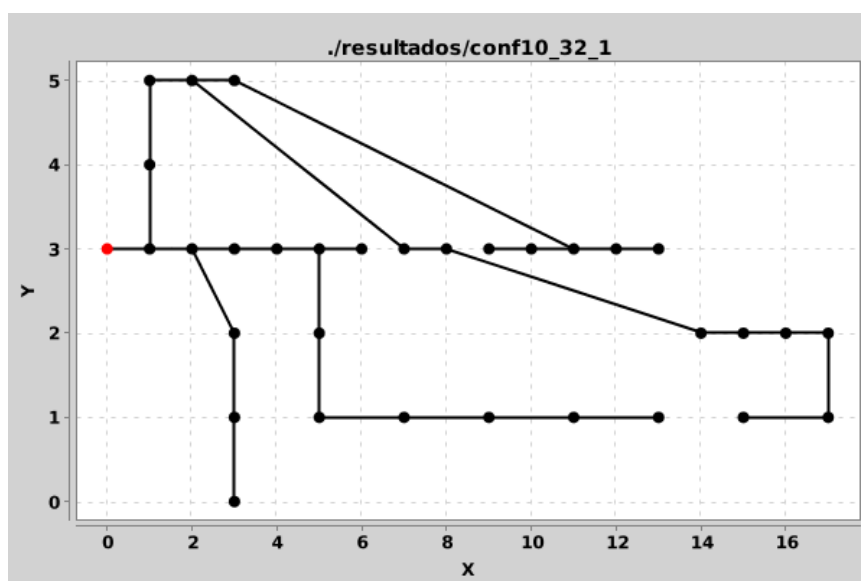


Figura 7: Rede reconfigurada bus\_32\_1.

## 4. Conclusão

Este projeto aplicou algoritmos genéticos ao problema da redução de perdas de energia em redes de distribuição através da reconfiguração de chaves, foram implementadas e testadas ao todo 10 configurações de GA, tendo testado 3 estratégias de seleção dos pais, 2 tipos de cruzamento, 2 taxas de mutação estática, mutação adaptativa, 3 maneiras de selecionar a nova população e a implementação de uma busca local.

Os melhores resultados foram alcançados através da utilização da busca local e da configuração com as estratégias mais promissoras para cada etapa do algoritmo. Sendo evidenciado o impacto positivo que a busca local teve sobre os resultados.

Com relação ao problema da redução de perdas em redes de distribuição foi comprovado que Algoritmos Genéticos são uma boa estratégia de solução, tendo apresentado uma redução de até 65% nas perdas de algumas redes.

Como trabalhos futuros pode-se listar a aplicação de outros algoritmos evolutivos, ou ainda de algoritmos de enxame e outras metaheurísticas ao problema da redução de perdas. Ainda se tratando de Algoritmos Genéticos existem outras estratégias evolutivas que não foram abordadas por esse trabalho e que merecem ser investigadas. Outro ponto interessante seria o aperfeiçoamento da busca local, uma busca local mais barata possibilitaria que mais iterações fossem realizadas no mesmo tempo o que pode levar a melhores resultados ou no caso das instâncias muito grandes como a instância bus\_1476\_84 o algoritmo poderia dar alguma resposta em tempo hábil.

## Referências

- [1] A. J. Monticelli and A. Garcia, *Introdução a sistemas de energia elétrica*. Campinas, SP: UNICAMP, 2<sup>a</sup> ed., 2011.
- [2] EPE: Empresa de Pesquisa Energética, *Balanço energético nacional 2018: ano base 2017*. Rio de Janeiro: Ministério de Minas e Energia, 2018.
- [3] SGT: Superintendência de Gestão Tarifária, “Tarifas/cálculo tarifário e metodologia/distribuição/perdas de energia,” 2018.
- [4] D. A. Lima, “Perdas de energia,” tech. rep., Superintendente de Gestão Tarifária, 2018.
- [5] L. M. O. de Queiroz, “Algoritmos genéticos híbridos para redução de perdas técnicas em redes primárias de distribuição considerando variações de demandas,” Master’s thesis, Universidade Estadual de Campinas (UNICAMP), Campinas, SP, 2005.
- [6] A. Merlin and H. Back, “Search for a minimal-loss operating spanning tree configuration in urban power distribution systems,” in *5th Power Systems Computation Conference*, (Cambridge, U.K), pp. 1–5, 1975.

- [7] M. E. Baran and F. F. Wu, “Network reconfiguration in distribution systems for loss reduction and load balancing,” *IEEE Transactions on Power delivery*, vol. 4, no. 2, pp. 1401–1407, 1989.
- [8] K. Nara, A. Shiose, M. Kitagawa, and T. Ishihara, “Implementation of genetic algorithm for distribution systems loss minimum re-configuration,” *IEEE Transactions on Power systems*, vol. 7, no. 3, pp. 1044–1051, 1992.
- [9] W.-M. Lin, F.-S. Cheng, and M.-T. Tsay, “Distribution feeder reconfiguration with refined genetic algorithm,” *IEE Proceedings-Generation, Transmission and Distribution*, vol. 147, no. 6, pp. 349–354, 2000.
- [10] Z. Ming, B. Pengxiang, L. Jian, and Z. Wenyuan, “An improved genetic algorithm for distribution system reconfiguration,” in *Proceedings. International Conference on Power System Technology*, vol. 3, pp. 1734–1738, IEEE, 2002.
- [11] M. F. N. d. Costa *et al.*, “Computação evolutiva para minimização de perdas resistivas em sistemas de distribuição de energia elétrica,” 1999.
- [12] E. M. Cavaleiro, A. H. Vergílio, and C. Lyra, “Optimal configuration of power distribution networks with variable renewable energy resources,” *Computers & Operations Research*, vol. 96, pp. 272–280, 2018.
- [13] M. Gendreau and J.-Y. Potvin, “Metaheuristics in combinatorial optimization,” *Annals of Operations Research*, vol. 140, pp. 189–213, Nov 2005.