

3 ALGORITMOS GENÉTICOS : CONCEITOS BÁSICOS E EXTENSÕES VINCULADAS AO PROBLEMA DE MINIMIZAÇÃO DE PERDAS

3.1 - Conceitos Básicos

Entendemos como *algoritmo* “um conjunto predeterminado e bem definido de regras e processos com operações finitas, destinados à solução de um problema com um número finito de etapas” e, como *genético*, “um termo relativo à genética, a qual é um ramo da biologia que estuda as leis da transmissão dos caracteres hereditários nos indivíduos, e os mecanismos que asseguram essa transmissão” (Ferreira, 1996).

Os *algoritmos* ditos *genéticos* são um tipo especial de algoritmo. Na realidade, são algoritmos de **busca estocástica polarizada** (Atmar, 1994). Portanto, uma definição de *algoritmos genéticos* poderia ser : um conjunto predeterminado e bem definido de regras e processos com operações finitas, destinados à **busca estocástica polarizada** da solução de um problema, com um número finito de etapas. Os algoritmos genéticos seguem as leis biológicas da transmissão dos caracteres hereditários nos indivíduos e os mecanismos que asseguram essa transmissão.

Apesar do caráter estocástico ou aleatório dos algoritmos genéticos, eles são capazes de explorar com eficiência a memória passada, no sentido de polarizar a busca por regiões promissoras no espaço de candidatos à solução (Goldberg, 1989).

Quanto à taxionomia, conforme pode ser observado na Figura 3.1, os algoritmos genéticos fazem parte dos sistemas computacionais inspirados em modelos de processos naturais, mais especificamente da computação evolutiva.

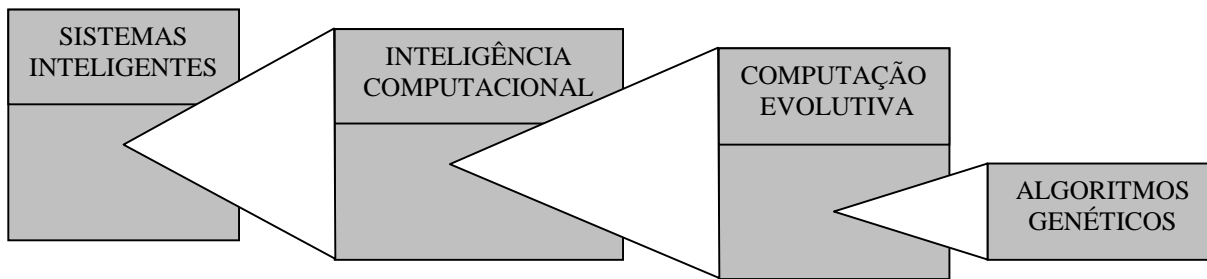


Figura 3.1 – Possibilidade de Classificação dos Algoritmos Genéticos

Na Figura 3.2, apresentamos de forma simplificada os passos básicos do processo de execução de um algoritmo genético (variações em relação a este esquema podem ser encontradas na literatura). Representamo-lo aqui como uma função que recebe como entrada uma *população inicial* e uma função de avaliação, normalmente denominada *fitness*, e fornece como saída o melhor *indivíduo* encontrado após o processo evolutivo. Para entendermos cada passo, necessitamos definir os seguintes termos : geração, população, indivíduo, função de fitness, seleção, reprodução e critério de parada.

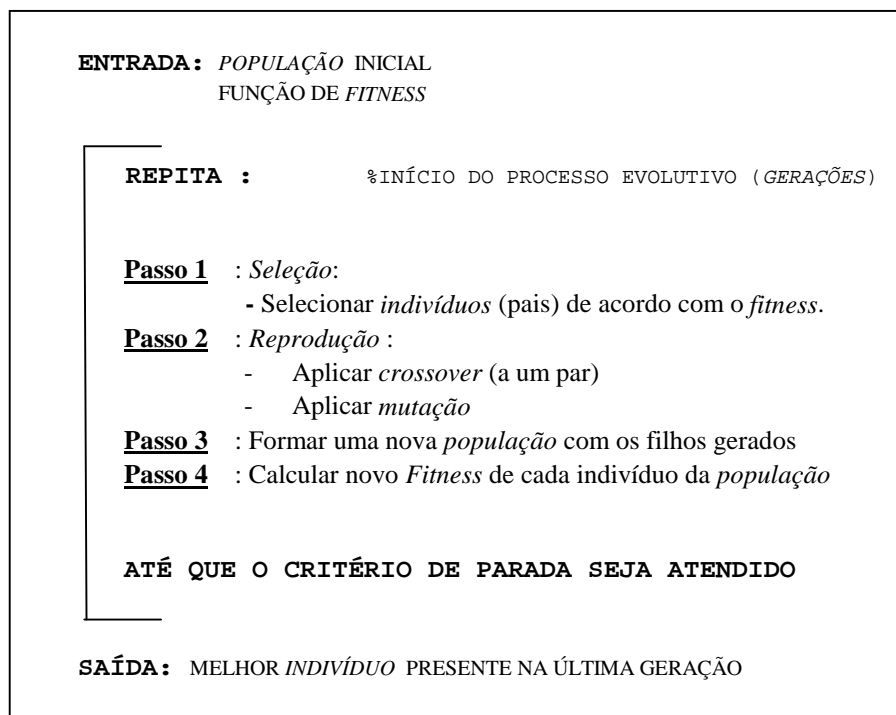


Figura 3.2 – Passos Básicos de um Algoritmo Genético

Geração

É o resultado da aplicação dos passos 1 a 4 junto a uma *população* (ver Figura 3.2)

População

Conjunto de *indivíduos* que compõe uma *geração*.

Indivíduo

Representação (genótipo) da solução (fenótipo) de um determinado problema (ambiente) através de um arranjo ou vetor cujos elementos são obtidos a partir de um alfabeto finito.

Função de Fitness

Função de avaliação, que mede o quão adaptado está o *indivíduo* ao ambiente.

Seleção

Processo de escolha que usualmente seleciona os *indivíduos* mais adaptados (que apresentam um maior *fitness*), para participarem do processo de *reprodução*.

Reprodução

Processo pelo qual *indivíduos* mais adaptados, ou seja, com melhores valores para a função de avaliação, têm a chance de perpetuar seu material genético, ou ao menos parte deste material, através da aplicação dos operadores genéticos de *crossover* e *mutação*.

Critério de Parada

Caracterizado pela obtenção de uma solução satisfatória ou pelo término do tempo computacional preestabelecido (Fogel, 1995).

Voltando à Figura 3.2 observa-se que o algoritmo genético parte de uma *população* inicial. Através de um mecanismo de *seleção*, dá preferência aos indivíduos mais adaptados ao ambiente (que apresentam maior *fitness*) e aplica os operadores genéticos de *crossover* e *mutação*. Caracteriza-se assim um processo evolutivo, onde espera-se que o *fitness* do melhor indivíduo a cada geração apresente uma tendência de crescimento continuado, sendo este indivíduo, por conseguinte, capaz de representar uma boa solução ao problema proposto, a qual pode não ser a melhor possível.

Como os algoritmos genéticos baseiam-se principalmente nos processos evolutivos das espécies, especialmente no mecanismo de seleção natural, descritos originalmente em Darwin (1859;1993), mostramos na Tabela 3.1 a correspondência entre alguns termos da linguagem adotada em ambiente computacional e na biologia.

Tabela 3.1 – Correspondência entre o Vocabulário Biológico e o Computacional

Termos Biológicos	Termos Computacionais
Cromossomo	Indivíduo
Gene	Caractere
Alelo	Valor do caractere
Lócus	Posição do caractere
Genótipo	Vetor de caracteres que representa o indivíduo
Fenótipo	Interpretação do vetor de caracteres

Procuramos detalhar nas próximas seções alguns tipos possíveis de representação, a dimensão das populações durante o processo evolutivo, certos mecanismos de seleção, as funções de avaliação ou *fitness* e determinados operadores genéticos. Em seguida, descrevemos em um estudo de caso, a implementação do algoritmo genético aplicado ao problema de

minimização de perdas em sistemas de distribuição de energia elétrica, apresentando resultados e comentários.

3.2 - Tipos de Representação

Usando termos da Tabela 3.1, cada indivíduo é considerado um “cromossomo” e, na representação clássica de algoritmos genéticos, ele é codificado como um vetor de caracteres de tamanho fixo, formado pela concatenação dos caracteres “0” e “1” do alfabeto binário {0,1} (representação binária). Existem variações para esta codificação, como a representação em ponto flutuante, onde o vetor de tamanho fixo é formado pela concatenação de elementos que são números reais (Michalewicz, 1996).

Representação Binária

No caso de uma representação binária, a codificação é direta quando as possíveis soluções já são de natureza binária. Caso contrário, procura-se transformar as possíveis soluções para o problema em um número binário ou em uma sequência de números binários.

Como exemplo, temos o problema de maximização da função $f(x) = \sin(x) + \sqrt{x}$, representada graficamente na Figura 3.3, onde x assume valores inteiros no intervalo [0,15]. Cada indivíduo representa a codificação binária de um valor inteiro da variável x no intervalo considerado. Como seus valores devem variar entre 0 e 15, cada indivíduo (candidato à solução) deverá possuir 4 bits. Em virtude da natureza do problema, a própria função $f(x)$ pode ser utilizada como função de *fitness* neste problema particular.

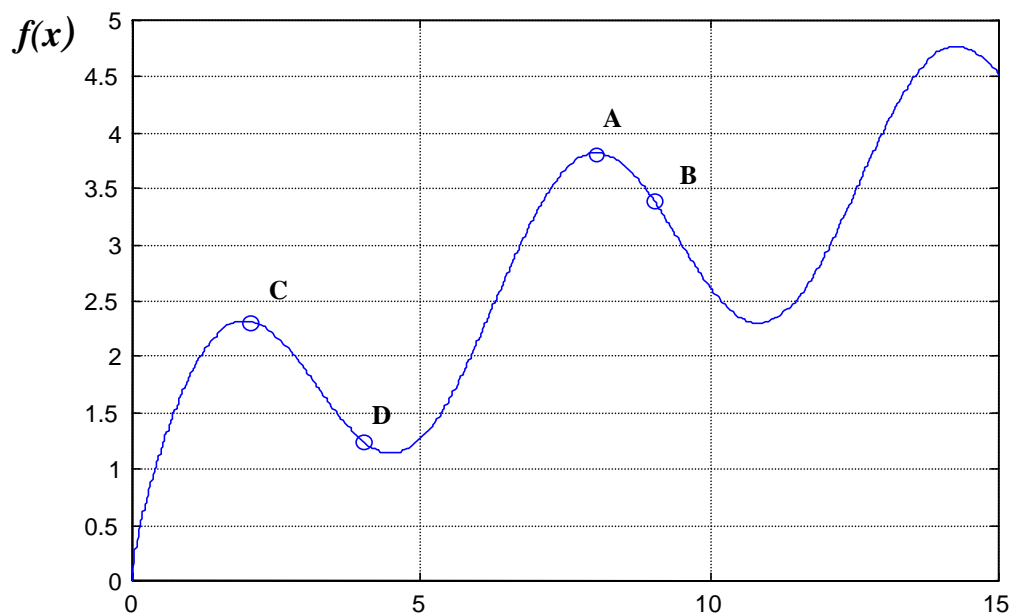


Figura 3.3 – Gráfico da Função a ser Maximizada, $f(x) = \text{sen}(x) + \sqrt{x}$

Observe na Tabela 3.2, uma população inicial gerada aleatoriamente para este problema de otimização, com 4 indivíduos.

Tabela 3.2 – Exemplo de uma População de Indivíduos com Codificação Binária

Candidato à solução	Codificação (genótipo)	x (fenótipo)	Fitness = $f(x)$
A	1 0 0 0	8	3,8178
B	1 0 0 1	9	3,4121
C	0 0 1 0	2	2,3235
D	0 1 0 0	4	1,2432

A partir desta população inicial (2ª coluna da Tabela 3.2), com base nos valores de fitness calculados para cada indivíduo (4ª coluna da Tabela 3.2), a aplicação repetida dos passos

apresentados na Figura 3.2 vai permitir evoluir a população, de geração em geração, até se atender o critério de parada. O melhor indivíduo que pode ser obtido neste caso é 1110, que corresponde a $x = 14$ e $f(x) = 4,7323$.

Representação em Ponto Flutuante

Na representação em ponto flutuante cada cromossomo é um vetor de tamanho fixo formado pela concatenação de números do conjunto dos números reais. Como exemplo, temos a representação adotada para aplicação junto ao problema de minimização de perdas, o qual veremos com detalhes na Seção 3.7.

Dependendo do problema, a representação em ponto flutuante apresenta uma série de vantagens em relação à representação binária. Entre elas podemos citar:

- o tamanho menor do cromossomo a ser considerado, reduzindo o consumo de CPU e memória no processamento computacional,
- a simplicidade da representação, tornando desnecessária a decodificação,
- maior controle sobre o efeito da perturbação causada pela aplicação do operador de mutação (preservação de critérios de vizinhança).

No entanto, a representação em ponto flutuante não permite o uso direto do conceito de blocos construtivos (building blocks) (Holland, 1975), conceito normalmente utilizado para demonstração de convergência do processo evolutivo.

3.3 - Tamanho da População

Um aspecto importante a ser considerado para o sucesso do processo evolutivo refere-se ao tamanho da população inicial e da população nas subseqüentes gerações, pois é possível comprometer a evolução dos indivíduos caso a quantidade escolhida seja muito grande ou muito

pequena. Esta medida, exceto em casos muito particulares, deve ser estabelecida empiricamente e de acordo com a disponibilidade de recursos computacionais (Goldberg, 1983 e 1989; Holland, 1975; Richards, 1995; Costa, 1999).

Deve-se considerar também a possibilidade da população ser dividida em subpopulações e aplicar-se diferentes formas de reprodução e de operadores para cada uma delas.

3.4 - Mecanismos de Seleção

Um outra etapa de grande impacto no desempenho do algoritmo genético é a fase de seleção de indivíduos para reprodução. Existem vários tipos de seleção que podem ser aplicados (Goldberg, 1989). Entre os mecanismos existentes destacamos o “Roulette Wheel”, adotado em Goldberg (1983) e a seleção por torneio.

A seleção por torneio é recomendada quando há uma grande diferença relativa no valor do fitness dos indivíduos. Em sua versão mais simples, para $k = 2$, um par de indivíduos é escolhido aleatoriamente e o indivíduo que possuir maior *fitness* será selecionado para reprodução. O processo se repete obedecendo a probabilidade de seleção normalmente estabelecida. O valor do parâmetro k está diretamente vinculado ao nível de pressão seletiva durante este processo (Michalewicz, 1996). Estudos comparativos entre este mecanismo e outros são apresentados em Koza (1996).

No mecanismo “Roulette Wheel” de seleção, a cada indivíduo é associada a porção de uma roleta, proporcional ao valor de seu fitness. Observe no exemplo da Figura 3.4 a formulação e aplicação desse mecanismo. Neste caso, após “rodar” a roleta, o indivíduo A foi o escolhido, apesar de possuir apenas 37,5 % de chance.

Conforme expusemos na definição de seleção, usualmente procura-se os indivíduos mais adaptados. Entretanto, com o objetivo de manter-se a diversidade da população durante a seleção, pode-se, algumas vezes, optar por indivíduos menos adaptados.

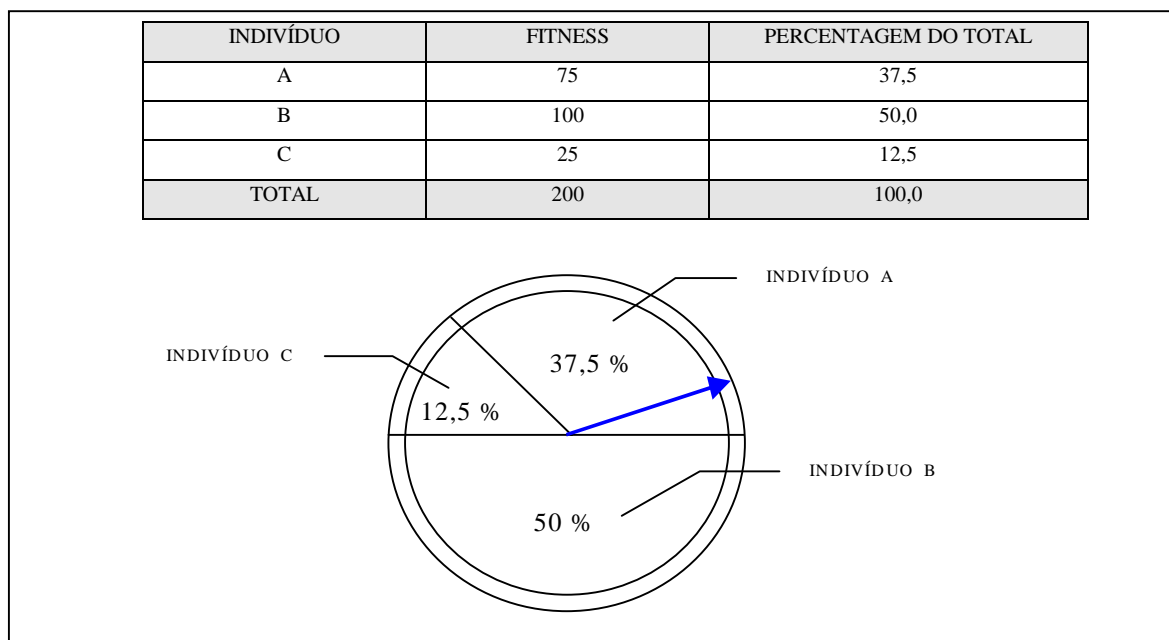


Figura 3.4 – Mecanismo “Roulette Wheel” de Seleção

Como já mencionado anteriormente, a escolha de um mecanismo de seleção apropriado é de suma importância durante o processo evolutivo. No entanto, exceto em casos particulares, a escolha é feita empiricamente, analisando-se as várias propostas existentes e o problema a ser abordado, não sendo ainda possível caracterizar a priori o melhor mecanismo.

3.5 - Função de Fitness

Para cada indivíduo, calcula-se a função de avaliação (*fitness*). Este valor está associado a quão adaptado está o indivíduo ao ambiente com o qual ele interage. Em termos computacionais, esta medida indica o quão bem um dado candidato à solução (indivíduo) é capaz de resolver o problema (ambiente). É imprescindível a definição apropriada desta medida de adaptação, para que o processo evolutivo seja capaz de fornecer uma solução que atenda aos objetivos em questão (Goldberg, 1989).

3.6 - Operadores Genéticos

A forma de codificação adotada para representar cada indivíduo (ou cromossomo) determinará quais operadores genéticos serão aplicados e como serão utilizados os operadores escolhidos. Tais operadores possuem variações. O operador de crossover pode ser simples, multiponto, uniforme, ordenado (Goldberg, 1989), aritmético (Michalewicz, 1996), ou uma combinação deles. O mesmo acontece com o operador de mutação, o qual possui variantes e todas dependem do tipo de codificação adotada para se definir o cromossomo.

Dentre os vários tipos de crossover destacamos o crossover simples. Observe na Figura 3.5 a aplicação deste operador a um par de cromossomos. Basicamente, os pares trocam entre si o código genético a partir de um ponto de corte determinado aleatoriamente.

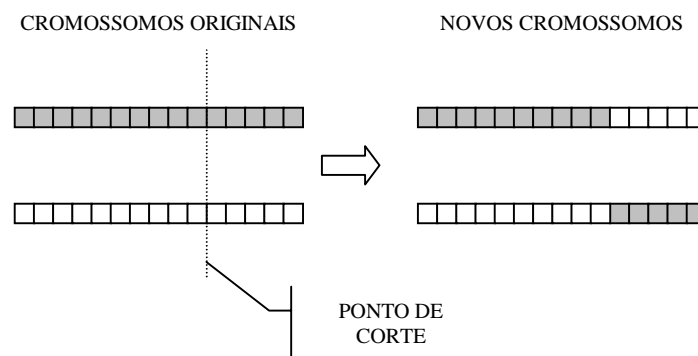


Figura 3.5 – Exemplo de Crossover Simples

O operador de mutação sempre opera sobre um único cromossomo, alterando seu material genético. Ele pode atuar em apenas um gene de cada vez (mutação simples) ou em uma porcentagem de genes determinada aleatoriamente (mutação aleatória). No caso de uma codificação binária, esta mutação ou alteração do código genético, corresponde simplesmente à

troca de um bit pelo seu complemento. No caso de uma codificação em ponto flutuante, a mutação corresponde ao acréscimo de um valor extraído de uma distribuição aleatória ao gene escolhido (ou aos genes).

Para representação em ponto flutuante, destacamos dois tipos de mutação : mutação uniforme, onde o valor aleatório a ser acrescido ao gene (ou genes) provém de uma distribuição constante ao longo do processo evolutivo; mutação não-uniforme, onde o valor a ser acrescido provém de uma distribuição que depende do tempo decorrido ao longo do processo evolutivo (Michalewicz, 1996).

3.7 - Algoritmo Genético Aplicado ao Problema de Minimização de Perdas

Introdução

Conforme exposto no Capítulo 2, o problema de minimização de perdas em sistemas de distribuição de energia elétrica caracteriza uma situação de otimização combinatória. A busca de soluções ótimas é um problema em que a complexidade computacional cresce exponencialmente com a dimensão da rede (Tão, 1997).

Este problema foi explorado por Costa (1999) a partir da aplicação de algoritmos genéticos com representação binária, juntamente com inovações na representação genética de estruturas em árvore. Com o objetivo de confirmar a necessidade de se encontrar soluções ótimas (estado das chaves seccionadoras) para cada perfil diferente de demanda, incorporamos representação genética em ponto flutuante ao algoritmo de solução para perfis de demanda fixa, proposto por Costa (1999).

A representação em ponto flutuante mostrou-se mais adequada, por preservar a noção de vizinhança na aplicação do operador de mutação, o que não ocorre com a representação binária.

Além disso, como apontado por Michalewicz (1996), esse tipo de codificação acelera o processo de evolução, é mais consistente de geração para geração e fornece uma melhor precisão, especialmente junto a problemas que requerem representação através de vetores de caracteres muito extensos.

Apresentamos na Figura 3.6 a rede de distribuição de energia elétrica considerada em todos os estudos de casos, neste capítulo e nos seguintes. Essa rede foi proposta inicialmente por Baran & Wu (1989). Ela é composta por 37 arcos, 32 nós e chaves seccionadoras associadas a cada arco. Na configuração apresentada, encontram-se abertas as chaves associadas aos arcos 33, 34, 35, 36 e 37.

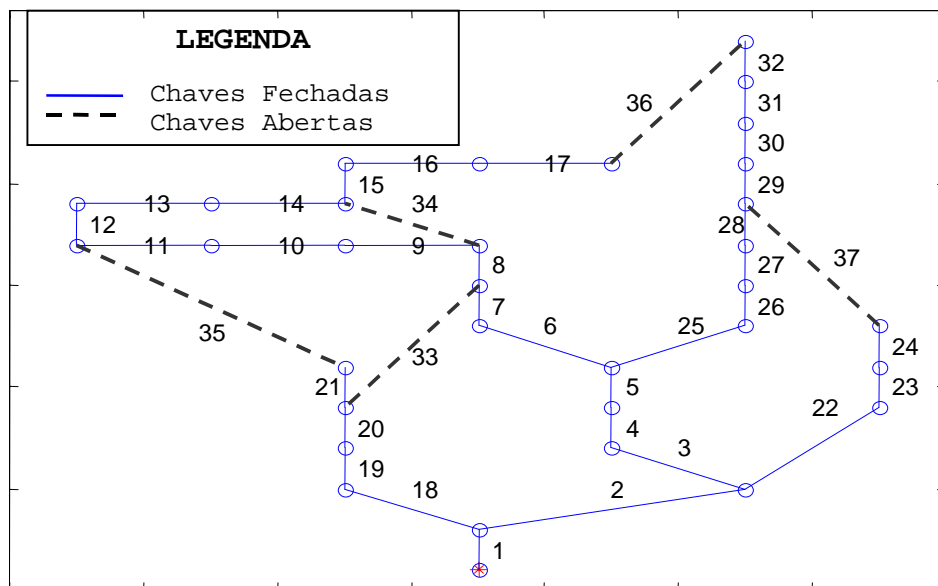


Figura 3.6 – Rede de Distribuição (Baran & Wu, 1989)

Objetivo

Para um dado perfil de demanda nos nós, buscamos minimizar as perdas de energia elétrica da rede de distribuição apresentada na Figura 3.6. Para tal, selecionamos seis propostas

de distribuição aleatórias de demanda para a rede, ilustradas na Figura 3.7. No eixo das ordenadas temos o valor da demanda para cada um dos 32 nós, os quais são representados no eixo das abscissas. Estes perfis são distintos mas possuem a mesma demanda total de energia.

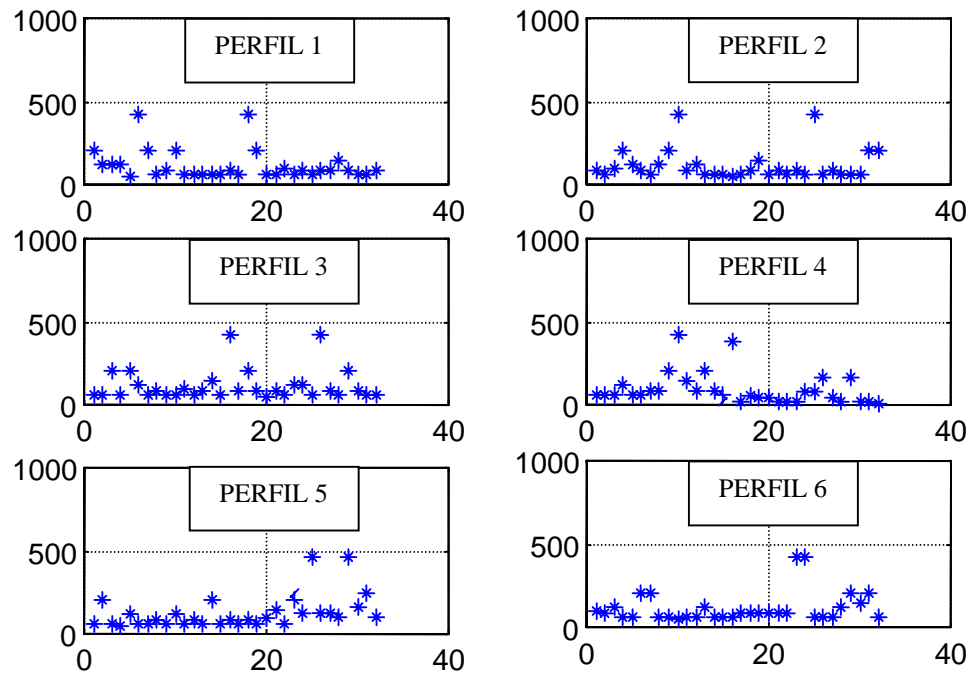


Figura 3.7 – Distribuição dos Perfis de Demanda

Descrevemos a seguir os principais componentes do algoritmo genético aplicado a este problema.

Representação

Como a solução do problema de perdas para a rede da Figura 3.6 resume-se em apontar as chaves seccionadoras que devem permanecer abertas, poderíamos simplesmente codificar indivíduos para indicar diretamente o estado das chaves : aberta ou fechada. Entretanto, devido às características do problema, com a exigência de soluções que mantenham a estrutura radial da

rede, impedindo a formação de ciclos e garantindo o suprimento de energia a todos os nós consumidores, não seria possível garantir, com esta codificação direta, a geração de filhos factíveis a partir de pais também factíveis.

Por esta razão, procuramos utilizar a mesma metodologia de obtenção de árvores geradoras de peso mínimo proposta por Costa (1999) e melhorar os resultados por ele obtidos. Assim, adotamos uma representação genética em ponto flutuante na definição dos dados de entrada para o algoritmo de obtenção da árvore geradora de custo mínimo, onde cada elemento do vetor de entrada de tamanho fixo, significa o peso ou custo de cada arco da rede, o qual é arbitrário e ainda não está associado ao perfil de demanda presente na rede de distribuição.

Como para cada vetor de entrada de tamanho fixo (indivíduo), sempre vai existir uma árvore geradora de custo mínimo associada (Ahuja *et al.*, 1993), obtemos, a partir desta árvore, a configuração proposta (ou estado das chaves seccionadoras) para cada indivíduo da população.

Basicamente, o indivíduo ou vetor de números reais no intervalo [0,1] (genótipo) deverá ser decodificado (gerando o fenótipo) para apontar quais chaves seccionadoras, do conjunto total de chaves da rede, devem ficar abertas. Observe nas Figuras 3.8 e 3.9 um exemplo do genótipo e do fenótipo correspondente de um indivíduo da população a ser evoluída.

INDIVÍDUO (GENÓTIPO)																		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0.15	0.28	0.78	0.29	0.74	0.02	0.35	0.45	0.26	0.90	0.48	0.64	0.82	0.19	0.76	0.39	0.91	0.05	0.59
20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	
0.27	0.43	0.60	0.56	0.93	0.23	0.78	0.39	0.83	0.71	0.49	0.14	0.42	0.62	0.46	0.71	0.42	0.86	
CUSTO OU PESO DO ARCO						NÚMERO DO ARCO												

Figura 3.8 – Exemplo do genótipo de um indivíduo gerado aleatoriamente. O custo ou peso dos arcos deverá ser evoluído para que a árvore geradora de custo mínimo obtida leve à minimização das perdas

INDIVÍDUO (FENÓTIPO)																		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	1	0	1	1	1	1	1	1	0	1	1	0	1	1	1	0	1	1
20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	
1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	

1 – FECHADO
0 – ABERTO

NÚMERO DO ARCO

Figura 3.9 – Exemplo do fenótipo correspondente ao genótipo do indivíduo da Figura 3.8. Esta configuração de chaves foi extraída da árvore geradora de custo mínimo, mas como os custos dos arcos da Figura 3.8, usados para obter a árvore, podem ainda não estar associados ao perfil de demanda presente na rede de distribuição, pode não haver minimização das perdas na rede de distribuição com a adoção desta configuração de chaves

A decodificação do fenótipo do indivíduo da Figura 3.9 indica que as chaves 3, 10, 13, 17 e 24 devem estar abertas.

Teremos um vetor com tantos números em ponto flutuante quantos forem os arcos do sistema. Como já mencionado anteriormente, a decodificação do vetor de números (indivíduo) segue a metodologia apresentada em Costa (1999). Costa utilizou o algoritmo de Prim para produzir uma árvore geradora de custo mínimo (condição necessária para o processo de evolução da população), a partir de um conjunto de arcos com custo ou peso fixo. Em nosso caso, adotamos o algoritmo de Sollin (Apêndice B), mais indicado para a geração de árvores esparsas, conforme sugerido por Ahuja *et al.*(1993) – situação típica das redes de distribuição de energia elétrica.

Dimensão das Populações

A população inicial é gerada aleatoriamente com 50 indivíduos (ou vetores) contendo 37 posições, denominadas *locus*, formadas pela concatenação de números reais no intervalo [0,1]. A dimensão da população inicial e subseqüentes é mantida constante durante o processo evolutivo.

Mecanismo de seleção

Foi adotado o mecanismo de seleção “Roulette Wheel”, apresentado na Seção 3.4.

Função de Avaliação

A função de avaliação em nossa abordagem está vinculada à função objetivo apresentada na Equação 2.1, Seção 2.3, Capítulo 2. Para uma dada configuração i ($G'=[N,A']$), de operação radial da rede de distribuição, o total de perdas é dado na forma

$Perdas(conf_i) = \sum_{k: a_k \in A'=Ac} r_k x_k^2$ e cada indivíduo será avaliado de acordo com o cálculo apresentado na Equação 3.1 :

$$Fit(i) = \frac{1}{10 * Perdas(conf_i)} \quad (3.1)$$

onde $Fit(i)$ representa o fitness do i -ésimo indivíduo durante o processo evolutivo e $Perdas(conf_i)$ representa o total das perdas associadas à configuração i . Como podemos observar, o fitness é inversamente proporcional às perdas do sistema, calculadas a partir da configuração $conf_i$, associada ao fenótipo do indivíduo i .

Por exemplo, o indivíduo i representado nas Figuras 3.8 e 3.9, possui um fitness de 1,6524 ($Fit(i) = 1,6524$), referente a uma perda de 0,06051877 ($Perdas(conf_i) = 0,06051877 p.u.$). Essas perdas correspondem a 605,1877 kW.

Operadores Genéticos

Aplicamos os operadores genéticos de crossover simples e mutação uniforme. Na aplicação da mutação uniforme, procuramos respeitar o intervalo de variação dos números reais

[0,1] de cada *locus* dentro do vetor (seqüência de custos nos arcos da rede), ajustando os valores para 0 ou 1 em caso de extrapolação dos limites.

Parâmetros

Os parâmetros adotados, após exaustivos testes de desempenho, estão apresentados na Tabela 3.3. Na 1^a coluna, temos o tamanho da população, igual a 50 indivíduos criados aleatoriamente. Na 2^a coluna, relatamos que 10% da população é selecionada por mecanismo “Roulette Wheel” (discutido na Seção 3.4) para reprodução a cada geração. Nas 3^a e 4^a colunas temos respectivamente a percentagem de aplicação de crossover e mutação, aplicados aos 10% de indivíduos escolhidos para reprodução.

Tabela 3.3 – Parâmetros Adotados

Tamanho da população	Percentagem da população selecionada para reprodução	Probabilidade de Crossover	Probabilidade de Mutação
50	10 %	90 %	1 %

3.8 - Resultados e Comentários

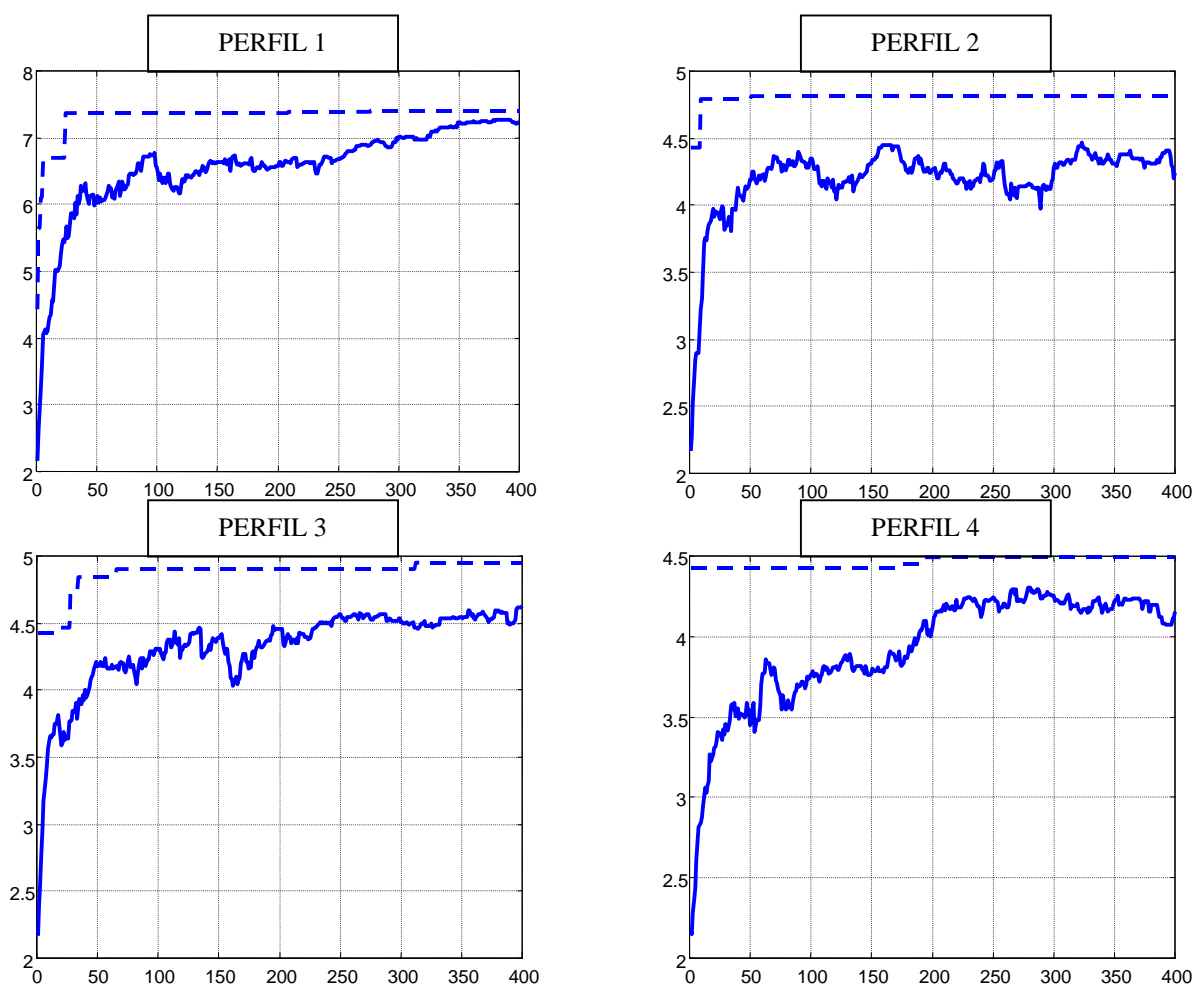
Na Tabela 3.4 apresentamos a estatística do desempenho do algoritmo genético aplicado a cada perfil de demanda ilustrado na Figura 3.7. Todas as simulações partiram da mesma população inicial gerada aleatoriamente e o cálculo da redução de perdas para todos os perfis toma como referência uma mesma configuração para a rede, chaves abertas : 33, 34, 35, 36 e 37. Essa configuração foi proposta por Baran & Wu (1989) – ela foi adotada neste trabalho para permitir comparações com outras abordagens. Observe na coluna 2 da Tabela 3.4, o valor da perda calculada em cada perfil para esta configuração. Na coluna 4, temos o valor das perdas

calculadas para cada perfil, considerando a configuração proposta pelo fenótipo do melhor indivíduo encontrado.

Tabela 3.4 – Estatística

PERFIL	PERDA REFERENCIAL (em kW)	FENÓTIPO DO MELHOR INDIVÍDUO (chaves abertas)	PERDAS CALCULADAS PARA O MELHOR INDIVÍDUO (em kW)	% DE REDUÇÃO NAS PERDAS
1	216,1733	7-10-14-16-28	135,0730	37,51 %
2	375,1484	7-10-14-17-28	207,7695	44,61 %
3	372,8436	7-9-14-17-28	202,0010	45,82 %
4	453,2261	10-14-15-28-33	222,4486	50,91 %
5	303,7213	7-9-14-28-36	178,0502	41,37 %
6	282,6649	7-9-14-32-37	204,1293	27,78 %

Na Figura 3.10 apresentamos, através da função de *fitness*, o desempenho do melhor indivíduo e a média do *fitness* da população.



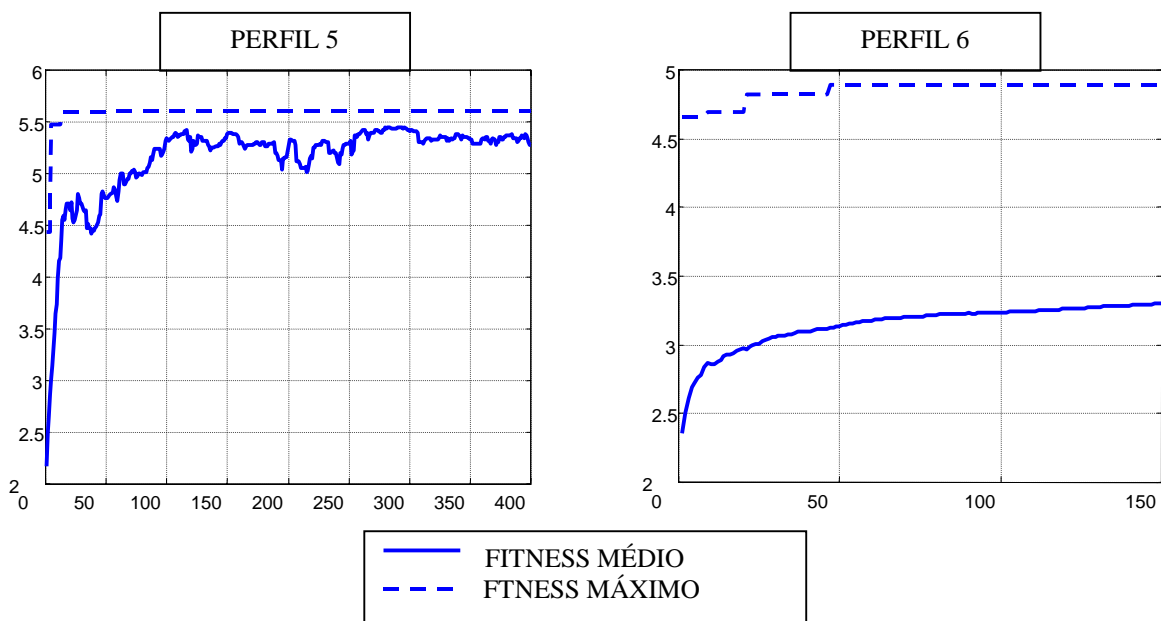


Figura 3.10 – Fitness Máximo e Médio da População para Cada Perfil de Demanda

Este estudo de casos, com aplicação de algoritmos genéticos para perfis distintos de demanda fixa, confirmou a necessidade de mudanças de configuração na rede na presença de demandas variáveis, procurando sempre manter a rede de distribuição em uma configuração de perdas mínimas. Isso pode ser observado a partir dos cálculos apresentados na Tabela 3.5.

Tabela 3.5 – Aumento das Perdas quando Diferentes Configurações são Aplicadas a Diferentes Perfis de Consumo

	P ₁	P ₂	P ₃	P ₄	P ₅	P ₆
P ₁	0	0,2911	2,2553	0,3059	1,7839	6,4485
P ₂	1,7181	0	1,0184	12,602	0,4144	4,2648
P ₃	1,5189	2,5179	0	13,277	0,1080	4,1104
P ₄	2,7524	2,8187	12,481	0	9,4363	12,819
P ₅	3,8348	2,4171	3,9084	16,925	0	1,5433
P ₆	17,918	32,532	21,187	26,369	17,757	0

A Tabela 3.5 ilustra o aumento das perdas, em percentagem, de uma determinada configuração ótima, encontrada para um perfil de consumo P_i ($i=1,...,6$), quando aplicada a outro perfil P_j ($j=1,...,6, j \neq i$). Por exemplo, a aplicação da melhor configuração encontrada para

o perfil P_1 (chaves abertas : 7, 10, 14, 16 e 28) em P_6 , resultou em um acréscimo de 6,4485 % nas perdas em relação ao valor determinado pela melhor configuração encontrada para P_6 (chaves abertas : 7, 9, 14, 32 e 37).

Para a situação de perfis de demandas variáveis, recorreremos a uma outra ferramenta da Computação Evolutiva, os Sistemas Classificadores. Esta ferramenta apresenta grande potencial de adaptação a “ambientes” variantes no tempo e que requerem ações contínuas e muitas vezes em tempo real (Booker *et al.*, 1989).

Antes de investigarmos o caso de perfil de demanda variável em cada nó do grafo, verificaremos o desempenho do Sistema Classificador para perfis de demanda fixa. Em seguida, realizaremos experimentos correspondendo a situações em que o perfil de demanda do sistema sofre variações abruptas (em degrau), representando praticamente o reinício do processo de geração de classificadores - pois os melhores classificadores encontrados para o primeiro perfil, possivelmente não apresentarão bom desempenho na resposta ao segundo perfil de demanda, requerendo uma reavaliação do conjunto de classificadores.

Finalmente, estudaremos variações suaves e periódicas nos perfis de demandas, mais condizentes com a prática. O propósito é obter um Sistema Classificador suficientemente flexível para indicar novas soluções sempre que as variações de demanda começarem a interferir significativamente no nível de perdas associado a uma determinada configuração da rede de distribuição (estado das chaves).

São abordados no Capítulo 4 os conceitos de Sistemas Classificadores. No Capítulo 5 detalhamos a implementação e resultados da aplicação dessa ferramenta ao problema de minimização de perdas em redes de distribuição de energia, com perfis de demandas fixas e variáveis em cada nó consumidor.