

4 TÉCNICA DE COMPUTAÇÃO EVOLUCIONÁRIA

4.1 ALGORITMO GENÉTICO

Algoritmos Genéticos foram desenvolvidos por John Holland e colegas da Universidade de Michigan [64]. Os objetivos da pesquisa eram, então, o estudo do funcionamento de processos adaptativos em sistemas naturais e a criação de programas de computador com as características dos mecanismos naturais investigados. Dentro deste contexto, a aquisição de sistemas robustos tornou-se um dos temas centrais da pesquisa em algoritmos genéticos. Caso os sistemas artificiais, hardware e software, desenvolvidos por cientistas, viessem a apresentar certas propriedades de sistemas naturais, tais como reprodução e auto-reparo, o alto custo associado à tarefa de reprojeter estes sistemas seria eliminado [21, 22, 65-67].

Entretanto, é na área de otimização que algoritmos genéticos têm encontrado a sua principal fonte de aplicação, particularmente naqueles problemas em que o tamanho e/ou complexidade do espaço de busca torna inviável a utilização de técnicas convencionais [68].

A literatura identifica três classes de métodos de busca ou otimização, aqueles baseados em cálculo, aqueles baseados em enumeração e os aleatórios. Os métodos baseados em cálculo baseiam-se, em geral, na computação do gradiente da função a ser otimizada. Os métodos de enumeração, que amostram todos os pontos do espaço de busca, são ineficientes quando não utilizados em conjunto com técnicas que fazem uso de localidade. Métodos de busca aleatória empregam, em geral, o princípio de geração de um ponto aleatório, comparação com outro gerado previamente, e a retenção do melhor. Estes métodos são a melhor solução apenas para problemas em que o espaço de busca é extremamente grande e complexo.

Por outro lado, Algoritmos Genéticos ou AGs diferem das técnicas descritas acima devido ao fato de eles trabalharem com uma população de soluções e de usarem regras de transição probabilísticas. Além disso, AGs não

usam, em princípio, nenhuma informação auxiliar sobre o espaço de busca, tal como o emprego de derivadas [21].

Através da herança de algumas das características dos algoritmos evolutivos descritos anteriormente, o funcionamento dos algoritmos genéticos pode ser explicado em termos da representação do problema; do uso de três operadores, seleção, crossover e mutação; e da aplicação de uma função de avaliação de aptidão.

A representação do problema diz respeito ao mapeamento das possíveis soluções presentes no espaço de busca em uma estrutura de dados que possa ser manipulada computacionalmente. Tipicamente, algoritmos genéticos codificam possíveis soluções em palavras binárias. Enquanto que estas palavras binárias recebem também a denominação de cromossomos ou genótipos, o ponto do espaço de busca codificado pelos mesmos recebe o nome de fenótipo [21, 22]. Desta forma, a implementação computacional de um AG inicia-se pela geração aleatória de uma população de genótipos. A partir da aplicação dos operadores genéticos, sucessivas gerações de indivíduos são então produzidas.

O operador de seleção baseia-se no princípio de sobrevivência dos mais aptos ou seleção natural observado na evolução biológica. A medida de desempenho de um indivíduo em relação a uma determinada especificação determinará a probabilidade de este indivíduo ser selecionado e contribuir para a criação de indivíduos em uma próxima geração. A característica mais importante do operador de seleção é o fato de ele ser probabilístico, o que diferencia os algoritmos genéticos dos algoritmos evolutivos descritos anteriormente.

Após o processo de seleção, pares de indivíduos são escolhidos de forma aleatória para execução do operador de recombinação ou crossover. Assim como no caso de estratégias evolutivas, o operador de crossover executa uma troca de informações entre dois indivíduos, a partir da hibridização do conteúdo de seus respectivos cromossomos. A partir do cruzamento de dois indivíduos via crossover, dois descendentes são produzidos, ambos

apresentando material genético dos dois progenitores. A aplicação ou não do crossover após a escolha de dois cromossomos é probabilística, isto é, dois novos híbridos podem ser produzidos, ou então os dois indivíduos inicialmente selecionados são preservados. A taxa de aplicação do operador de crossover é em geral alta ($> 60\%$) [21].

Após a aplicação do operador de crossover, a nova geração de indivíduos passa pelo processo de mutação. Este operador é aplicado, com uma determinada taxa de probabilidade, a cada bit dos cromossomos. A taxa de aplicação deste operador é usualmente muito baixa, sendo em geral empregada com menos de 1% de probabilidade por bit.

A literatura clássica de algoritmos genéticos considera o operador de crossover como sendo o principal no mecanismo de funcionamento de AGs, ao passo que o operador de mutação teria apenas um caráter secundário [21]. Entretanto, muitos pesquisadores da área apresentam uma opinião contrária, acreditando que o operador de mutação é o mais importante no direcionamento de um algoritmo genético.

Conforme mencionado anteriormente, a probabilidade de seleção de um indivíduo é determinada pela sua medida de desempenho, também chamada de aptidão. A avaliação dos indivíduos é feita através da aplicação de uma função de aptidão, que é definida com base na especificação do problema. Tipicamente, trata-se de uma função a ser otimizada. Após a aplicação desta função, cada indivíduo tem associado a si um valor escalar, real ou inteiro, que determina o quão apto aquele indivíduo é em relação à especificação do problema. Conforme será visto posteriormente, estes valores podem ser normalizados, para evitar possíveis problemas na etapa de seleção. A Figura 11 resume o mecanismo básico de funcionamento de AGs.

O critério de parada, definido no fluxo da Figura 11, pode ser um de convergência de indivíduos (em termos de genótipos ou de aptidão), um número máximo de gerações produzidas, e/ou a parada no momento em que um indivíduo com uma aptidão satisfatória seja encontrado.

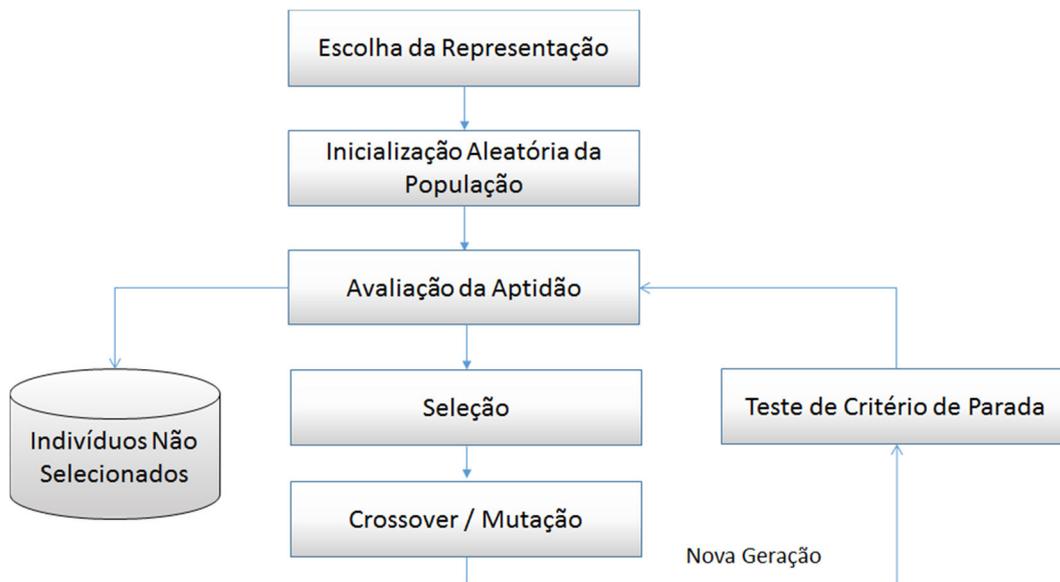


Figura 11 – Fluxo básico de funcionamento de um algoritmo genético com três operadores

4.1.1 Representação

A representação é um aspecto fundamental na modelagem de um algoritmo genético para a solução de um problema. Neste estágio define-se a estrutura do cromossomo, com os respectivos genes que o compõem, de maneira que este seja capaz de descrever todo o espaço de busca relevante do problema.

4.1.2 Codificação e Decodificação

A solução de um problema pode ser representada por um conjunto de parâmetros (genes), unidos para formar uma cadeia de valores (cromossomo); a este processo chama-se codificação. As soluções (cromossomos) são

codificadas através de uma sequência formada por símbolos de um sistema alfabético.

Originalmente, utilizou-se o alfabeto binário (0, 1), porém, novos modelos de AGs codificam as soluções com outros alfabetos, como, por exemplo, com números reais [23].

A decodificação do cromossomo consiste basicamente na construção da solução real do problema a partir do cromossomo. O processo de decodificação constrói a solução para que esta seja avaliada pelo problema.

4.1.3 Avaliação

A avaliação permite ao algoritmo genético determinar sua proximidade à solução do problema. Ela é feita através de uma função que melhor representa o problema e tem por objetivo oferecer uma medida de aptidão de cada indivíduo na população corrente, que irá dirigir o processo de busca. Dado um cromossomo, a função de avaliação consiste em associar-se um valor numérico de “adaptação”, o qual se supõe proporcional à sua "utilidade" ou "habilidade" do indivíduo representado em solucionar o problema em questão.

4.1.4 Operadores Genéticos

Os operadores mais conhecidos nos algoritmos genéticos são os de Reprodução, Cruzamento (Crossover) e Mutação.

Reprodução: refere-se ao processo de selecionar e copiar um determinado cromossomo para a população seguinte de acordo com sua aptidão. Isto significa que os cromossomos mais aptos, têm maior probabilidade de contribuir para a formação de um ou mais indivíduos da população seguinte. Existem basicamente os seguintes métodos: troca de toda população, troca de toda população com elitismo, onde todos os cromossomos

são substituídos, sendo o cromossomo mais apto da população corrente copiado para população seguinte, e troca parcial da população (*steady state*), onde os M melhores indivíduos da população corrente são copiados para população seguinte [21-23].

Cruzamento: é um operador baseado na troca de partes dos cromossomos (pais), formando-se duas novas soluções (filhos). Este processo pode ser observado no exemplo a seguir na Figura 12, onde a solução está codificada com alfabeto binário.

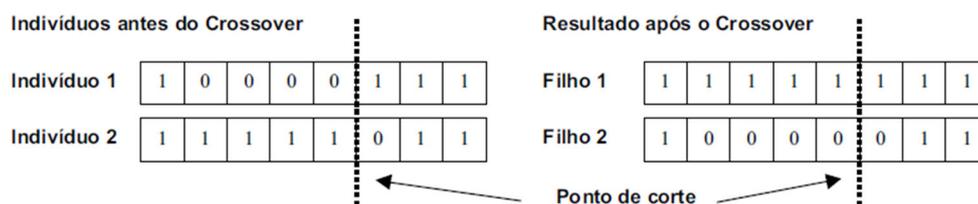


Figura 12 – Cruzamento de um único ponto

O ponto onde ocorre o corte para a realização do cruzamento é escolhido aleatoriamente; no exemplo da Figura 12 utilizou-se um único ponto, mas podem ser realizados cortes em mais de um ponto, caracterizando o *multi-point crossover* [21, 23, 64]. Para realizar o cruzamento, primeiro é necessária a escolha, por sorteio, dos cromossomos “pais”. Em seguida ocorre a realização ou não do cruzamento segundo um parâmetro, denominado taxa de cruzamento. Deste modo, de acordo com a taxa de cruzamento, pode ocorrer que os cromossomos “pais” sejam repassados sem modificação para a geração seguinte, criando “filhos” idênticos a eles. A ideia do operador de Cruzamento é tirar vantagem (*exploit*) do material genético presente na população.

Mutação: é a troca aleatória do valor contido nos genes de um cromossomo por outro valor válido do alfabeto. No caso de alfabeto binário troca-se de 0 para 1 e vice-versa. Da mesma forma que para o cruzamento, utiliza-se uma taxa de mutação que, para cada bit da sequência de caracteres,

sorteia-se se ocorrerá ou não a mutação; no caso de ocorrência, o bit será trocado por outro valor válido pertencente ao alfabeto Figura 13.

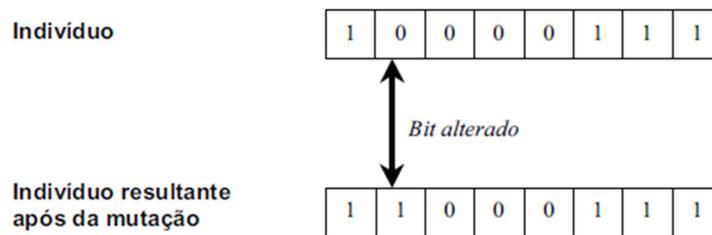


Figura 13 – Mutação

A mutação garante a diversidade das características dos indivíduos da população e permite que sejam introduzidas informações que não estiveram presentes em nenhum dos indivíduos. Além disto, proporciona uma busca aleatória (*exploration*) no AG, oferecendo oportunidade para que mais pontos do espaço de busca sejam avaliados.

4.1.5 Parâmetros da Evolução

Os parâmetros que mais influenciam o desempenho do algoritmo genético são:

Tamanho da População: afeta o desempenho global e a eficiência dos Algoritmos Genéticos. Uma população muito pequena oferece uma pequena cobertura do espaço de busca, causando uma queda no desempenho. Uma população suficientemente grande fornece uma melhor cobertura do domínio do problema e previne a convergência prematura para soluções locais. Entretanto, com uma grande população tornam-se necessários recursos computacionais maiores, ou um tempo maior de processamento do problema. Logo, deve-se buscar um ponto de equilíbrio no que diz respeito ao tamanho escolhido para a população.

Taxa de Cruzamento: probabilidade de um indivíduo ser recombinado com outro. Quanto maior for esta taxa, mais rapidamente novas estruturas serão introduzidas na população. Entretanto, isto pode gerar um efeito indesejável, pois a maior parte da população será substituída, causando assim perda de variedade genética, podendo ocorrer perda de estruturas de alta aptidão e convergência a uma população com indivíduos extremamente parecidos, indivíduos estes de solução boa ou não. Com um valor baixo, o algoritmo pode-se tornar muito lento para oferecer uma resposta aceitável.

Taxa de Mutação: probabilidade do conteúdo de um gene do cromossomo ser alterado. A taxa de mutação previne que uma dada população fique estagnada em um valor, além de possibilitar que se chegue a qualquer ponto do espaço de busca. Porém deve-se evitar uma taxa de mutação muito alta, uma vez que esta pode tornar a busca essencialmente aleatória, prejudicando fortemente a convergência para uma solução ótima.

Intervalo de Geração: controla a porcentagem da população que será substituída durante a próxima geração (substituição total, substituição com elitismo, substituição dos piores indivíduos da população atual, substituição parcial da população sem duplicatas). Esse número de indivíduos substituídos também é conhecido como GAP.

Número de gerações: representa o número total de ciclos de evolução de um Algoritmo Genético, sendo este um dos critérios de parada do algoritmo genético. Um número de gerações muito pequeno causa uma queda no desempenho: um valor grande faz necessário um tempo maior de processamento, mas fornece uma melhor cobertura do domínio do problema, evitando a convergência para soluções locais.