

ALGORITMOS GENÉTICOS E OTIMIZAÇÃO

JOSÉ RICARDO POTIER DE OLIVEIRA

APOSENTADO PELO INSTITUTO DE PESQUISAS DA MARINHA, BRASIL

jricpotier@gmail.com

INTRODUÇÃO

Otimização é um conhecido ramo da matemática cujo objetivo é encontrar valores de parâmetros, que definem uma função que modela um problema, e de modo a que esta função atinja o seu valor máximo, ou mínimo, dependendo da natureza do problema. A otimização é muito útil em diferentes campos de atuação humana, como economia, transportes, logística, produção e vários outros. O clássico problema do caixeiro viajante é um exemplo típico de aplicação de otimização. Neste caso, tenta-se achar a melhor rota para um caixeiro viajante percorrer um certo número de cidades, sem passar duas vezes pela mesma, e voltando ao ponto de partida. Problemas de otimização vão-se tornando cada vez mais complexos à medida que se aumenta a complexidade da função objetivo, bem como o número de parâmetros, a verosimilhança da não linearidade da função objetivo e a existência de restrições no problema. Para tais casos o uso de Algoritmos Genéticos (AGs) tem levado a resultados excelentes.

Os AGs foram introduzidos por John H. Holland [1] e David E. Goldberg [2], e constituem métodos de busca inspirados nos mecanismos naturais de seleção, adaptação e evolução de populações de seres vivos, procurando atingir uma solução ótima para o problema proposto.

Numa breve (e incompleta) taxonomia dos métodos de otimização podemos classificá-los em métodos determinísticos e métodos probabilísticos. Dentro dos métodos probabilísticos encontramos os Algoritmos Evolutivos ou Evolucionários, estando os AGs nesta última classe.

Algoritmos Genéticos são uma poderosa ferramenta na abordagem de problemas complexos de otimização, envolvendo muitas variáveis, parâmetros e restrições.

A INSPIRAÇÃO NA NATUREZA

Na Natureza os indivíduos com melhor adaptação ao ambiente são, geralmente, os que sobrevivem por mais tempo (*seleção natural*), conseguindo reproduzir-se. A reprodução entre dois indivíduos que sobreviveram à seleção natural faz com que o material genético de ambos se misture, gerando descendentes com características de ambos os progenitores. Esta mistura (ou *cruzamento*) entre melhores indivíduos traz a tendência para que as novas gerações apresentem, cada vez mais, indivíduos mais bem adaptados do que os de gerações anteriores. De forma esporádica, com uma frequência muito pequena, uma *mutação* genética pode ocorrer em poucos indivíduos de uma geração para outra. Esta mutação pode levar a um salto na evolução, produzindo um indivíduo superior, com mais chances de sobreviver à seleção natural, e que vai transmitir as suas características genéticas à próxima geração. Também pode ocorrer que a mutação produza um indivíduo pior, que muito provavelmente será eliminado na etapa da seleção natural. Desta forma, repetindo por séculos as etapas de seleção, cruzamento e mutação, a Natureza promove uma constante melhoria na capacidade de adaptação e sobrevivência dos indivíduos de uma espécie.

DA INSPIRAÇÃO BIOLÓGICA PARA O ALGORITMO

O AG básico vai adotar a mesma estratégia evolutiva que a mãe Natureza, e vai trabalhar com operadores *seleção*, *cruzamento* e *mutação*. Partimos de um conjunto inicial de possíveis “soluções” (ou soluções-candidatas) do nosso problema, geradas aleatoriamente, que representam uma população inicial, sendo cada uma destas soluções-candidatas um indivíduo da população inicial. Cada um destes indivíduos será representado por uma estrutura de dados, que será denominada de *cromossoma*. Para exemplificar, vamos considerar um problema simples (que obviamente não necessita de ser resolvido por AG), como o de encontrar o mínimo da função

$$f(x) = x^4 - \frac{13}{2}x^3 + \frac{203}{16}x^2 - \frac{257}{32}x + \frac{85}{16}.$$

Vamos admitir, *a priori*, que a solução x_{min} se encontra no intervalo $[0, 4]$, e que pretendemos que a mesma tenha precisão de três casas decimais. Assim, teremos 4001 valores que podem ser a solução procurada. É claro que este é um problema simples, e testar todas as possibilidades não levaria muito tempo para um computador. Mas este exemplo serve para outro propósito: mostrar como repre-

sentar cada indivíduo da população, para trabalhar mais facilmente as etapas seguintes: cruzamento e mutação. Vamos associar a cada valor do conjunto onde sabemos que está a solução do problema, com a precisão requerida, uma palavra binária com 12 bits ($2^{12} = 4096$), cobrindo toda a faixa de valores. Assim, os valores binários vão representar diretamente as possibilidades de solução, pois $000000000000=0$ e $111111111111=4095$. Seguidamente geremos aleatoriamente uma população inicial com seis indivíduos (correspondentes a seis valores de x). Deve-se destacar que, dependendo do problema, muitas outras formas de representação binária são possíveis (e mesmo não binária). O gráfico da função e a população inicial podem ser vistos na figura 1. A representação binária de cada um dos seis indivíduos da população inicial são os valores de x dos pontos no gráfico, e equivalem a um cromossoma na comparação com o processo biológico.

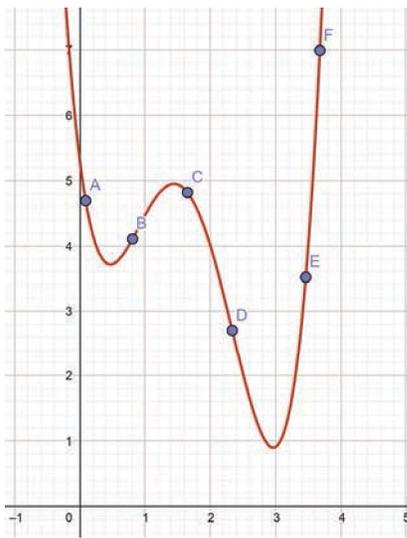


Figura 1. Gráfico da função objetivo e população inicial.

SELEÇÃO

Neste ponto, precisamos de estabelecer a “aptidão” de cada indivíduo em resolver o problema. Neste exemplo simples é claro que os indivíduos mais “aptos” são os que fornecem os menores valores de $f(x)$, já que procuramos o seu mínimo. Então, a própria $f(x)$ também nos serve como função aptidão (função objetivo). No entanto, em problemas mais complexos, estabelecer uma função aptidão adequada é um passo importante que depende da habilidade em representar o problema.

Observando-se o gráfico, podemos ver que os três menores valores de $f(x)$ são dados pelos pontos B, D e E. O processo seguinte é a seleção destes três indivíduos e o descarte de A, C e F, por serem os menos aptos. Na prática, não conhecemos a forma da função objetivo, e são técnicas e algoritmos típicos em AGs que realizarão a seleção (torneio, roleta, etc.). No entanto, estes não serão apresentados aqui, pois é mais apropriado fazê-los num curso sobre AGs, devido à extensão do assunto. Sendo assim, vamos apenas admitir que algum algoritmo selecionou as três melhores soluções entre as disponíveis na população inicial.

CRUZAMENTO

O próximo passo é fazer com que os indivíduos mais aptos se cruzem entre si, para criar uma nova geração de indivíduos que, teoricamente, podem ser mais aptos. Esta etapa é simples quando utilizamos cromossomas binários: sorteia-se uma posição de corte para um par de cromossomas (*pais*) e trocamos as suas partes, como demonstrado na Figura 2. Desta forma, podemos, por exemplo, cruzar B com D, B com E e D com E, sorteando pontos de corte para cada par de cromossomas e gerando três novos indivíduos (*filhos*) para substituir os eliminados.

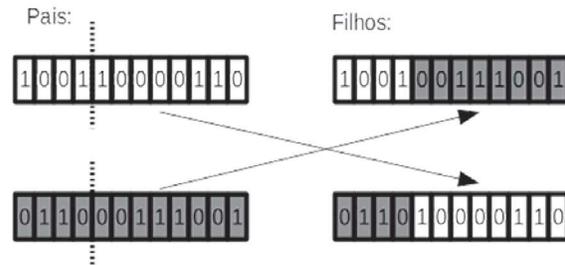


Figura 2. Operação Cruzamento.

MUTAÇÃO

O passo seguinte é realizar a mutação de alguns bits, como mostrado na figura 3, nos indivíduos da população, escolhidos aleatoriamente com uma pequena probabilidade (tipicamente entre 0,1% e 5%). Assim, no nosso exemplo, com seis indivíduos com 12 bits cada, teremos $12 \times 6 = 72$ bits. Se usarmos uma probabilidade de 4% de ocorrência de mutação, teremos aproximadamente 3 bits ($0,04 \times 72 = 2,88$), que terão os seus valores invertidos, considerando toda a população.

Após este passo, voltamos aos passos de avaliação desta população, seleção dos indivíduos mais aptos, cru-

zamento e mutação, até que algum critério de término seja satisfeito. Resumidamente, o AG básico é representado como:

- ▶ Início
- ▶ Iniciar população
- ▶ **Avaliar** indivíduos da população
- ▶ Repetir
 - ▶ **Selecionar** indivíduos para reprodução
 - ▶ Executar **cruzamento**
 - ▶ Executar **mutação**
 - ▶ **Avaliar** indivíduos da população
- ▶ Até
- ▶ Critério de parada satisfeito
- ▶ Fim

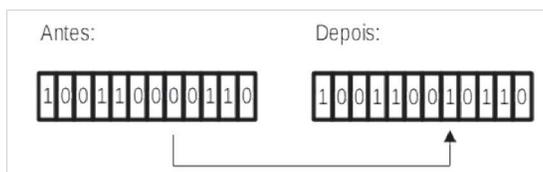


Figura 3. Operação Mutação.

CONSIDERAÇÕES PRÁTICAS SOBRE UM MODELO PARA ESTRATÉGIAS DE COMBATE

A definição de uma função objetivo que avaliará as soluções-candidatas, a representação dos dados do problema em forma de cromossomas, bem como possíveis estratégias e algoritmos para executar as operações de seleção, cruzamento e mutação são parte essencial da aplicação de AGs, e podem ter várias formas. Novamente, tais matérias não serão detalhadas aqui, por ser mais apropriado fazê-lo num curso mais completo sobre AGs. Aqui faremos apenas a ilustração de alguns aspetos práticos através de um exemplo que é apresentado com detalhe em [3]. O problema consiste em criar a simulação de uma estratégia (solução) de combate ótima, de uma força naval aliada contra outra inimiga. A força inimiga vai apresentar um número arbitrário e fixo de navios e seus armamentos, e a estratégia de combate é definir quantos e quais navios da frota aliada enviar para combate. A complexidade do problema consiste no facto de que cada força possui um conjunto de armamentos: a frota aliada possui até nove navios, com 1 a 5 armas por navio, cada arma com uma probabi-

lidade de acertar o alvo em função da distância, um custo financeiro por arma efetivamente utilizada, e um risco envolvido, devido à capacidade de reação da frota adversária. A função objetivo, MoE , (do inglês *Measure of Effectiveness*) foi idealizada refletindo estas três considerações e a solução ótima é encontrar a combinação de quantos navios, e que armas utilizar e a que distância permanecer. Existe um cálculo de a probabilidade do navio acertar o seu alvo em função da distância a que ele se encontra, denominado $MoE_{AcProbDamage}$, cálculo adaptado de [4], que deve ser maximizado. Uma segunda parte da função objetivo é denominada MoE_{AcCost} , a qual estima o custo financeiro relativo de cada arma usada na estratégia (que deve ser minimizada), e uma terceira parte avalia o risco que representa aquela estratégia, MoE_{Risk} , devido à capacidade de reação do inimigo (que deve ser minimizada). A função objetivo a ser maximizada é, então, definida como:

$$MoE = MoE_{AcProbDamage} + (1 - MoE_{AcCost}) + (1 - MoE_{Risk}).$$

Neste problema, a população inicial era de 100 indivíduos. Cada possível solução (cromossoma) para o problema considera um número diferente de navios da frota aliada, com seus armamentos, e a diferentes distâncias da frota inimiga. Essa forma de representação leva-nos a ter cromossomas de tamanho variável, uma interessante possibilidade dos AGs que demonstra a sua versatilidade na busca de soluções, como representado na figura 4, em que é mostrada a forma de executar o cruzamento. Evidentemente são necessários algoritmos que impeçam algumas combinações que não podem ser permitidas.

A mutação, nesta forma de representação, pode ser simplesmente a troca de um navio do cromossoma por outro que não está presente.

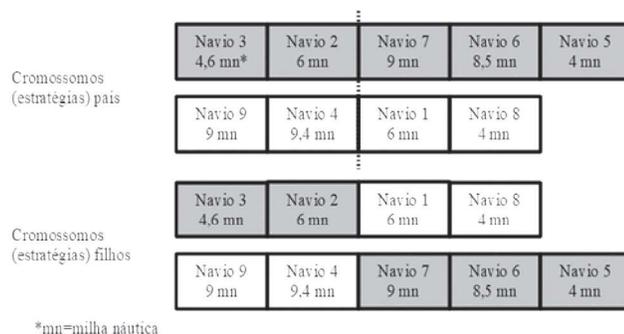


Figura 4. Cruzamento implementado em [3].

Tabela 1. Soluções em [3]

Navio	Cenário 1		Cenário 2		Cenário 3	
	Selecionado	Distância	Selecionado	Distância	Selecionado	Distância
1	Sim	0,3 mn	Não	-	Não	-
2	Sim	1,8 mn	Não	-	Não	-
3	Sim	0,3 mn	Sim	13,87 mn	Sim	33,83 mn
4	Sim	0,24 mn	Sim	0,22 mn	Sim	0,06 mn
5	Sim	0,15 mn	Sim	0,76 mn	Sim	0,09 mn
6	Sim	0,05 mn	Sim	0,49 mn	Sim	1,62 mn
7	Sim	0,15 mn	Sim	0,73 mn	Sim	0,2 mn
8	Sim	1,31 mn	Sim	0,19 mn	Sim	1,06 mn
9	Sim	1,98 mn	Sim	0,17 mn	Não	-

Os testes realizados em [3] consideraram três cenários, sempre com a frota aliada disposta de até nove navios, com as características descritas anteriormente. No cenário 1, a frota inimiga apresenta-se com uma frota com poderio igual ao total da frota aliada. O resultado do algoritmo determina o envio da frota aliada completa, especificando as distâncias a serem mantidas. No cenário 2, a frota inimiga apresenta-se com quatro dos seus navios com maior poderio. No último cenário, a frota inimiga apresenta-se também com quatro navios, porém de menor poderio bélico. Os resultados são sintetizados na tabela 1.

Um especialista militar foi convidado a resolver os mesmos cenários, e os resultados foram compatíveis com os obtidos com o AG ([3]).

INTERPRETAÇÃO DO FUNCIONAMENTO E VULNERABILIDADES DOS AGS

A função aptidão projetada para o problema representa uma superfície em \mathbb{R}^n , repleta de vales e picos, a partir da qual se procura encontrar um pico (ou vale, dependendo do problema) que seja uma solução satisfatória. Satisfatória, pois nem sempre é ótima, uma vez que existem muitos máximos (ou mínimos) locais nesta superfície, e o AG não nos garante que o máximo (ou mínimo) global foi atingido.

Em problemas mais complexos, com muitas variáveis e parâmetros, com características não lineares, restrições, objetivos múltiplos, grandes espaços de busca, etc., a visualização de gráficos como o da figura 1 não é possível, devido à complexidade e à dimensionalidade da superfície em \mathbb{R}^n , e técnicas matemáticas clássicas são difíceis de ser utilizadas. Nestes casos, o emprego do AG tem-

-se mostrado muito eficaz na resolução destes problemas.

Os indivíduos de cada geração são pontos pertencentes a esta superfície, espalhados aleatoriamente por ela, e que serão avaliados quanto ao seu grau de validade como solução (função aptidão). A seleção elimina pontos que não apresentam resultados satisfatórios e mantém os outros para a etapa de cruzamento, quando se espera que a combinação de dois indivíduos selecionados venha a produzir outro que esteja mais próximo do resultado procurado (máximo ou mínimo). A mutação representa um salto aleatório na superfície de busca, que pode levar a regiões até então não exploradas da mesma. Tais características colocam os AGs como métodos de busca probabilística, porém com uma ação direcionada, não caracterizando uma procura completamente aleatória.

É preciso, no entanto, ressaltar alguns factos quanto à utilização dos AGs e problemas relacionados: a escolha do tamanho da população, das taxas de mutação e cruzamento, do ponto de corte dos cromossomos pais, e outras características de projeto dos AGs são escolhas que seguem muito mais o empirismo e a intuição do projetista do que regras determinísticas. Más escolhas podem levar o algoritmo a problemas de convergência (lentidão ou convergência prematura sem atingir resultado satisfatório). O tamanho da população, por exemplo, deve ser compatível com a superfície de busca. Poucos indivíduos podem não ser representativos do espaço de busca, e o algoritmo pode convergir para um mínimo local não satisfatório. Por outro lado, muitos indivíduos podem tornar o algoritmo lento. Taxas altas de mutação podem levar a perda de bons indivíduos, taxas baixas podem deixar re-

giões da superfície de busca sem serem exploradas. Em suma, a utilização de AGs pode ter de passar por experiências e ajustes até se encontrar a solução satisfatória, mas ainda assim é um método bastante eficaz na solução de problemas complexos de otimização.

REFERÊNCIAS

[1] Holland, J. H. *Adaptation in Natural and Artificial Systems*. MIT Press, 1975.

[2] Goldberg, D. E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[3] Rangel, P., Oliveira, J. R. Potier, Carvalho, J. G., Lima, B. S. L. P., Guimarães, S. *A Fuzzy Evolutionary Simulation Model (FESModel) for Fleet Combat Strategies*. GECCO 2013. Proceedings of the 15th annual conference companion on Genetic and evolutionary computation, 2013.

[4] Wagner, D. H. *Naval Operations Analysis*. Naval Institute Press, 1999.

SOBRE O AUTOR

José Ricardo Potier de Oliveira é engenheiro eletrônico, e possui mestrado em Processamento de Sinais pela COPPE/UFRJ. É aposentado pelo Instituto de Pesquisas da Marinha, órgão da Marinha do Brasil, onde trabalhou com o desenvolvimento de sistemas navais e processamento de radar. Foi também professor da Universidade Santa Úrsula e da Faculdade Cenequista da Ilha do Governador, no Rio de Janeiro, Brasil.



LOJA
spm

Consulte o catálogo e faça a sua encomenda online em www.spm.pt