

Teste diagnóstico - Algoritmo Genético, problema do caixeiro viajante

Luis Eduardo Misquita Freitas

Ciências da Computação – Universidade Federal do Tocantins (UFT)

1. Introdução

Este trabalho consiste no desenvolvimento de um algoritmo genético para resolver um determinado problema. O algoritmo foi desenvolvido utilizando a linguagem de programação Python. O problema do vendedor é um exemplo clássico de problema de otimização híbrida. Queremos saber qual o melhor caminho que o vendedor/caixeiro pode fazer sem ficar repetindo a cidade-estado.

2. Codificação

No problema, tratamos de cidades ou lugares e para facilitar a visualização e interpretação, optamos por usar o alfabeto numérico. Cada cidade foi associada a um número correspondente. Por exemplo, se tivéssemos N cidades, com N igual a 25, a primeira cidade receberia o valor 0 e a última cidade receberia o valor 24 ou N-1. Exemplo do cromossomo:

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24]

3. Cromossomos

A quantidade de genes em cada indivíduo é atribuída de acordo com o número de cidades escolhidas pelo usuário, nos testes o valor era 25 cidades, logo o número de genes em um cromossomo é 25.

4. Função de avaliação

Para avaliação da população, é calculada a distância do alelo atual(n) com o próximo(n+1). Para calcular a distância foi usada a fórmula:

$$D_{ab} = (x_b - x_a)^2 + (y_b - y_a)^2$$
 a distância dos alelos x_a e x_b é armazenada em uma variável e somada com a distância dos próximos alelos x_{a+1} e x_{b+1} , até o último alelo do cromossomo.

4.1. Operador de seleção

Para selecionar os genes que os futuros descendentes irão receber, o tipo de operador de seleção escolhido foi o de amostragem direta, onde uma cidade e sua antecessora irão ser escolhidas para compartilhar os alelos.

5. Operador de cruzamento

No processo de cruzamento entre dois indivíduos, A e B, foi empregada uma técnica de randomização para selecionar dois valores dentro do indivíduo A. A diferença entre esses valores no vetor é então transmitida para a descendência. Para preencher as lacunas nos alelos ausentes, o indivíduo B contribui com os seus próprios alelos para o filho.

6. Mutação

A mutação ocorre através do método da roleta, onde só ocorre quando um valor randômico for abaixo da taxa de mutação, que foi definida em 0.1 ou 1%. Quando o indivíduo é escolhido para sofrer mutação, um de seus alelos irá ser escolhido randomicamente para poder mudar de valor com outro alelo.

7. Substituição

Após a conclusão do ciclo, é realizado um processo de análise para determinar quais indivíduos serão eliminados. Um indivíduo só será descartado se a sua distância não exceder o valor da melhor rota estabelecido no algoritmo.

8. Parâmetros genéticos

Os parâmetros escolhidos foram:

- Número de Cidades: 25
- Tamanho da População: 1000
- Taxa de Mutação: 0.1 ou 1%
- Número de Gerações: 2000

Lista de cidades:

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24]

A distância entre as cidades são respectivamente são, onde ler-se (x, y):

[(113, 188), (55, 131), (145, 197), (98, 107), (106, 31), (43, 180), (48, 67), (125,

189), (95, 55), (56, 171), (37, 194), (1, 94), (105, 101), (164, 18), (34, 102), (37, 178), (55, 160), (84, 60), (178, 28), (8, 33), (111, 153), (11, 14), (162, 136), (93, 8), (52, 122)]

O algoritmo para depois de uma geração gerada. Quando ocorre, todas as populações são ordenadas e é escolhida aquela que tem a menor distância percorrida.

9. Conclusão

O algoritmo é capaz de encontrar uma solução rápida para o problema do caixeiro viajante, mas para que ele seja eficiente, é necessário realizar um grande número de testes e aumentar o limite de gerações antes de atingir a condição de parada.

References

AVI TOMAR, Travelling Salesman Problem using Genetic Algorithm. Medium, 13 de ago. de 2019. Disponível em: <<https://medium.com/thecyphy/travelling-salesman-problem-using-genetic-algorithm-130ab957f165>>. Acesso em: 30 de jun. de 2021.