



Joice Cristina da Silva Santos

Problema do Caixeiro Viajante usando Algoritmos Genéticos



**Problema do caixeiro viajante:** O problema do caixeiro-viajante (*PCV*) é um problema que tenta determinar a menor rota para percorrer uma série de cidades (visitando uma única vez cada uma delas), retornando à cidade de origem. Ele é um problema de otimização NP-difícil inspirado na necessidade dos vendedores em realizar entregas em diversos locais (as cidades) percorrendo o menor caminho possível, reduzindo o tempo necessário para a viagem e os possíveis custos com transporte e combustível.

**Algoritmos Genéticos:** Os Algoritmos Genéticos (AGs) diferem dos métodos tradicionais de busca e otimização, principalmente em quatro aspectos:

1. AGs trabalham com uma codificação do conjunto de parâmetros e não com os próprios parâmetros.
2. AGs trabalham com uma população e não com um único ponto.
3. AGs utilizam informações de custo ou recompensa e não derivadas ou outro conhecimento auxiliar.
4. AGs utilizam regras de transição probabilísticas e não determinísticas.

Algoritmos Genéticos são muito eficientes para busca de soluções ótimas, ou aproximadamente ótimas em uma grande variedade de problemas, pois não impõem muitas das limitações encontradas nos métodos de busca tradicionais.

Além de ser uma estratégia de gerar-e-testar muito elegante, por serem baseados na evolução biológica, são capazes de identificar e explorar fatores ambientais e convergir para soluções ótimas, ou aproximadamente ótimas em níveis globais.

"Quanto melhor um indivíduo se adaptar ao seu meio ambiente, maior será sua chance de sobreviver e gerar descendentes": este é o conceito básico da evolução genética biológica. A área biológica mais proximamente ligada aos Algoritmos Genéticos é a Genética Populacional.

Funcionamento básico do algoritmo: Inicialmente, é gerada uma população formada por um conjunto aleatório de indivíduos que podem ser vistos como possíveis



soluções do problema. Durante o processo evolutivo, esta população é avaliada: para cada indivíduo é dada uma nota, ou índice, refletindo sua habilidade de adaptação a determinado ambiente. Uma porcentagem dos mais adaptados são mantidos, enquanto os outros são descartados (darwinismo). Os membros mantidos pela seleção podem sofrer modificações em suas características fundamentais através de mutações e cruzamento (crossover) ou recombinação genética gerando descendentes para a próxima geração. Este processo, chamado de reprodução, é repetido até que uma solução satisfatória seja encontrada.

### Implementação:

**Tamanho da população:** 10

**Mutação:** É definida pelo trecho abaixo, onde a probabilidade de aceitar um novo gene mutado é calculada. O valor da probabilidade é determinado pela diferença de aptidão entre o novo gene e o gene existente. A taxa de mutação é definida por **prob > 0.5**.

```
else:
    prob = pow(
        2.7,
        -1
        * [
            (float)(new_gnome.fitness - population[i].fitness)
            / temperature
        ],
    )
    if prob > 0.5:
        new_population.append(new_gnome)
        break
```

**Cruzamento:** É definida pelo trecho de código abaixo, onde um novo gene (new\_g) é gerado a partir de um gene existente (p1.gnome). Isso é feito na função mutatedGene, que faz uma troca aleatória de duas cidades no gene. Isso é uma forma de "cruzamento" que cria um novo indivíduo a partir de dois pais.



```
while True:
    new_g = mutatedGene(p1.gnome)
    new_gnome = individual()
    new_gnome.gnome = new_g
    new_gnome.fitness = cal_fitness(new_gnome.gnome, distance_matrix)

    if new_gnome.fitness <= population[i].fitness:
        new_population.append(new_gnome)
        break
```

#### Resultados Obtidos:

##### População Inicial:

##### Valor Fitness

0121213651412120 802  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802

Current temp: 10000

Geração 1

GNOME Valor Fitness

0121213651412120 802  
0121213611452120 773  
0121123651412120 769  
0123211651412120 753  
0121213651412120 802  
0121213651412120 802  
0121213651412120 802



0151213621412120 752  
0126213151412120 694  
0121613251412120 768

Current temp: 9000.0

Geração 2

GNOME Valor Fitness

0126212151412130 667  
0151213621412120 752  
0213211651412120 784  
0121613252412110 676  
0121143651212120 796  
0121413611252120 741  
0121213251416120 768  
0121213654112120 791  
0521213611412120 764  
0123211651412120 753

Current temp: 8100.0

Geração 3

GNOME Valor Fitness

0126212151412130 667  
0121615232412110 686  
0121413116252120 683  
0151213621112420 650  
0123411651212120 780  
0421213611512120 773  
0121213151426120 718  
0211211653412120 737  
0121213654111220 681  
0121143151212620 688

Current temp: 7290.0

Geração 4

GNOME Valor Fitness

0151213621112420 650  
0126214151212130 667  
0121211654131220 698  
0121413116252120 683  
0121615232112140 734  
0124113151212620 643  
0123211151426120 652  
0511211623412120 710



0426213111512120 640  
0121411651232120 753

Current temp: 6561.0

Geração 5

GNOME Valor Fitness

0421213161512120 790  
0124113151212620 643  
0121213621115420 690  
0122311151426120 575  
0126214351212110 616  
0121413216152120 768  
0621211154131220 648  
0511211223416120 675  
0121615232121140 734  
0121411651232210 645