

Resolução do problema 8 *queens* utilizando algoritmos genéticos

Larissa Daiane Caneppele Guder

Bacharelado em Sistemas de Informação - Faculdade Três de Maio - SETREM

Av. Santa Rosa - 2405 Três de Maio - RS- Brazil

lariguder10@gmail.com

1. O problema

O problema das oito rainhas é um quebra-cabeças baseado na peça Rainha do jogo de tabuleiro Xadrez. No jogo de Xadrez, a Rainha é a peça mais poderosa, pode mover em qualquer direção: colunas, fileiras e diagonais, sem limite de casas no seu movimento (conforme imagem). O objetivo do quebra-cabeças é posicionar 8 rainhas em um tabuleiro 8 x 8, de forma que nenhuma rainha ataque a outra. O ataque acontece quando duas ou mais rainhas se encontram na mesma coluna, fileiras ou diagonais. Um algoritmo para otimizar este problema, e eventualmente obter um valor ótimo, deve minimizar quantidade de ataques. O valor ótimo global é zero ataques

2. Resolução

A resolução do problema se dá através da utilização da linguagem de programação R, com algoritmos genéticos, através da permutação. Tendo em vista que o tabuleiro tradicional contém 64 posições (oito linhas por oito colunas). Para tal, o espaço de busca inicial é de 4.426.165.368. O problema possui apenas 12 soluções possíveis.

Portanto, a limitação inicial é de uma rainha por linha e uma rainha por coluna. Onde a solução sempre será representada da seguinte forma:

x1	x2	x3	x4	x5	x6	x7	x8	
[1,]	5	7	2	4	8	1	3	6

Para cada elemento do array são considerados seu índice, como sendo a posição na linha, e o valor existente no índice, sendo considerado o valor existente na coluna. Sempre serão apresentadas oito rainhas no tabuleiro.

A limitação do número de rainhas por coluna acontece ao limitar o número de repetições que o valor representativo da rainha em 1.

Para a validação das diagonais, a seguinte equação foi utilizada:

$$CR1 - CRx = LR1 - LRx$$

Onde CR1 representa a coluna da rainha selecionada, e CRx a coluna da próxima rainha. LR1 representa a linha da coluna selecionada e LRx a linha da próxima rainha a ser comparada. Se a diferença entre a posição da coluna for igual a diferença da posição das rainhas nas linhas, significa que existem duas rainhas em conflito.

Conforme presente na documentação do pacote GA, o tipo de algoritmo genético a ser executado depende da natureza da decisão variáveis. Os valores possíveis são:

1. "binary": para representações binárias de variáveis de decisão.
2. "real-value": para problemas de otimização onde as variáveis de decisão são representações de ponto flutuante de números reais.

3. "permutation": para problemas que envolvem reordenar uma lista de objetos

Tendo em vista o fato de que o problema consiste em ordenar as rainhas de forma que não haja conflito, foi adotado a utilização de permutação. Devido ao fato de sofrer mutações, nem em todas execuções o valor encontrado será o correto.

O valor "fitness" é calculado com base no número de colisões existentes. O número máximo de colisões que pode ocorrer em um tabuleiro 8x8 é 28. Estando na mesma linha ou coluna ou diagonal. A solução ideal teria 0 colisões, portanto, uma adequação de 28.

O estado atual a ser considerado é 28 (o pior possível). A função de adequação é quanto maior melhor. Então, para o cálculo é subtraído do valor máximo de colisões, o valor de colisões encontradas na iteração.

3.1. Melhor solução

Abaixo, o resultado da resolução do problema, utilizando permutação, com 10000 iterações.

Genetic Algorithm

GA settings:

Type = permutation
Population size = 50
Number of generations = 10000
Elitism = 2
Crossover probability = 0.8
Mutation probability = 0.3

GA results:

Iterations = 10000
Fitness function value = 28
Solutions =
x1 x2 x3 x4 x5 x6 x7 x8
[1,] 5 3 1 7 2 8 6 4
[2,] 6 4 7 1 8 2 5 3

Referências

Package 'GA'. Disponível: <https://cran.r-project.org/web/packages/GA/GA.pdf>.
Acesso: setembro / 2018.