

Inteligência Artificial

Hill Climbing e Simulated Annealing

Nesta tarefa você vai implementar os algoritmos Hill Climbing e Simulated Annealing e usá-los para resolver o problema das N -Rainhas. Os programas devem ser feitos preferencialmente na linguagem Python.

1. Modelagem

- (a) Descreva como um tabuleiro $N \times N$ com N rainhas é representado no seu programa.

2. Implementação:

As seguintes funções devem ser implementadas de maneira que possam ser testadas individualmente.

- (a) Defina uma função que dado o tamanho do tabuleiro N , retorna um tabuleiro $N \times N$ com N rainhas. O tabuleiro deve ser gerado de maneira aleatória.
- (b) Defina uma função que dado um tabuleiro qualquer, retorna todos os seus vizinhos.
- (c) Defina uma função que dado um tabuleiro qualquer, retorna um de seus vizinhos. A escolha do vizinho a ser retornado pela função deve ser aleatória.
- (d) Defina uma função que dado um tabuleiro qualquer, retorna a avaliação deste tabuleiro (número de ataques entre as rainhas).

3. Hill Climbing

- (a) Implemente uma versão do algoritmo Hill Climbing, onde o tabuleiro sucessor do tabuleiro corrente será o primeiro vizinho dele (tabuleiro corrente) que tem uma avaliação melhor. Assim, se a avaliação do tabuleiro corrente Tc for igual a k , o primeiro tabuleiro vizinho de Tc encontrado com avaliação menor que k deve passar a ser o novo tabuleiro corrente.
- (b) Implemente uma versão do algoritmo Hill Climbing, onde todos os tabuleiros vizinhos do tabuleiro corrente são avaliados e escolha para ser o novo tabuleiro corrente aquele que mais melhora a avaliação do tabuleiro corrente atual. No caso de haver mais de um tabuleiro, a escolha deve ser feita de forma aleatória.
- (c) Analise o desempenho de cada umas das implementações. Considere:
 - Tabuleiros de tamanho 4, 8, 16 e 32.
 - Para cada tamanho de tabuleiro:
 - Indique quantas vezes você precisou executar os programas para encontrar uma solução.
 - Quantos tabuleiros correntes, em média, foram gerados em cada execução dos programas.
- (d) Quais conclusões você consegue tirar destes experimentos?

4. Simulated Annealing

- (a) Implemente o algoritmo Simulated Annealing. Os parâmetros de entrada do programa devem ser *temperatura inicial* $TempInicial$, o *número máximo de iterações* $MaxIt$ e o *fator de decaimento* α .
- (b) Determine experimentalmente valores para os parâmetros de entrada que permitam que uma solução seja encontrada. Considere apenas tabuleiros de tamanho 4 e 8. Relate como a busca se comportou para os diferentes valores dos parâmetros de entrada que você usou.

- (c) Caso você tenha sido bem sucedido no item anterior (encontrou a solução do problema de 4 e 8 rainhas), use os mesmos parâmetros de entrada usados e execute o programa agora considerando tabuleiros de tamanho 16 e 32. O que acontece?
5. Comparando os métodos Hill Climbing e Simulated Annealing na resolução do problema das N -rainhas, o que você pode concluir?