

Trabalho I do Componente Curricular de Inteligência Artificial de 2019

Algoritmo Genético

Éverton de A. Vieira e Gabriel H. Moro

17 de maio de 2019

Introdução

Objetivo

O objetivo do trabalho é o desenvolvimento e uso de um algoritmo genético para a resolução de um problema a escolha.

Problema Escolhido

O problema escolhido é o problema de coloração de vértices de grafos. O problema consiste em, dado um grafo G qualquer, colorir seus vértices de forma que pares de vértices ligados por uma aresta não possuam a mesma cor.

Desenvolvimento

Para o desenvolvimento do trabalho foi utilizada a linguagem *Python 3.5+* e as bibliotecas *pyeasyga*, para o algoritmo genético, e *Networkx*, para geração e exibição dos grafos.

Funcionamento do Programa

O código recebe como entrada quatro valores inteiros, passados como parâmetro na linha de comando de execução, sendo o primeiro a quantidade de vértices, o segundo é a quantidade de vértices que será utilizado pelo algoritmo de geração do grafo para conectar os vértices do ciclo inicialmente gerado pelo

algoritmo, o terceiro a probabilidade do algoritmo mudar as ligações das arestas geradas inicialmente entre os vértices e o quarto argumento é a quantidade de gerações que o algoritmo deve executar.

Exemplo de comando:

```
python main.py n k p g
```

No exemplo acima é gerado um grafo com n vértices, e o algoritmo executará g gerações.

O grafo é gerado usando a função *connected_watts_strogatz_graph* da biblioteca Networkx, que cria um anel com n vértices para então criar arestas com k nodos mais próximos (ou $k - 1$ caso n seja ímpar). Então novas arestas são criadas seguindo que, para cada aresta (u, v) do anel gerado inicialmente existe a probabilidade p desta aresta ser trocada por uma nova aresta (u, w) com chance uniforme de escolha para o vértice w . O processo inteiro é repetido até o resultado ser um grafo conexo.