

Introdução à Inteligência Artificial

Licenciatura em Engenharia Informática, Engenharia Informática – Pós-Laboral e
Engenharia Informática – Curso Europeu

2º Ano – 1º semestre

2022/2023

Trabalho Prático nº 2 *Problema de Otimização*

1. Objetivos

Conceber, implementar e testar métodos de otimização que encontrem soluções de boa qualidade para diferentes instâncias do problema a seguir descrito.

2. Maximum Edge Subgraph Problem

Dado um grafo e um valor inteiro k , o *Maximum Edge Subgraph Problem* consiste em encontrar um subconjunto de k -vértices tal que o número de arestas dentro do subconjunto seja máximo.

Formalmente o problema é definido:

Dados

- um grafo não direcionado $G = (V, A)$, composto por um conjunto V de vértices ligados entre si por arestas A
- um inteiro k

Problema:

- encontrar um subconjunto de vértices S , de tamanho k , tal que $S \subseteq V$, de forma a maximizar o número de arestas desse subconjunto
- o objetivo deste problema é, portanto, de maximização.

Instâncias para teste

Estão disponíveis no Moodle algumas instâncias deste problema para testar os algoritmos. Os algoritmos a implementar devem estar preparados para lidar com grafos que contenham até 500 vértices. Os grafos estão armazenados nos ficheiros de acordo com a representação lista de adjacências.

A primeira linha contém o valor de k .

Depois disso surge uma linha iniciada por “*p edge*”, onde são indicados o número de vértices e o número de arestas.

Os vértices são identificados por números inteiros a partir de 1. Cada uma das linhas seguintes inicia-se com a letra e que especifica uma aresta entre dois vértices.

A seguir ilustra-se como é armazenada a informação nas instâncias de teste (ficheiros de texto disponibilizados no *Moodle*):

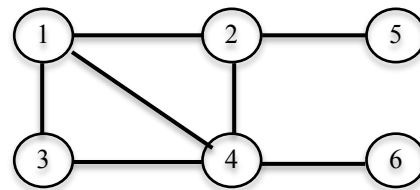
```
k 18
p edge 45 918
e 2 1
e 3 1
e 3 2
e 4 1
e 4 2
...
e 45 39
e 45 40
e 45 41
e 45 42
```

Valor de k
 Grafo com 45 vértices e 918 arestas
 Primeira aresta liga vértices 1 e 2
 Segunda aresta liga vértices 1 e 3
 ...
 Última aresta liga vértices 42 e 45

Exemplo:

Imagine que tem um grafo V com 6 vértices e 7 arestas, com a informação representada no ficheiro abaixo (*teste.txt* do Moodle). A imagem à direita ilustra o grafo. Pretende-se encontrar uma solução do problema para $k = 4$.

```
k 4
p edge 6 7
e 1 2
e 1 3
e 1 4
e 2 4
e 2 5
e 3 4
e 4 6
```



Para $k = 4$, considere três soluções possíveis (haverá outras):

$S1 = \{1, 2, 5, 6\}$ – número de arestas = 2

$S2 = \{1, 2, 3, 4\}$ – número de arestas = 5

$S3 = \{2, 3, 5, 6\}$ – número de arestas = 1

Como se trata de um problema de maximização, a solução **S2** será a melhor das três acima apresentadas.

3. Métodos de Otimização

Neste trabalho pretende-se que implemente e avalie a capacidade de diferentes algoritmos de otimização para encontrar soluções de boa qualidade para o problema descrito. Sendo assim, no seu estudo deverá implementar os 3 métodos seguintes e efetuar um estudo comparativo aprofundado sobre o desempenho da otimização. No Moodle estão disponibilizados vários ficheiros de instâncias, com complexidades diferentes, para efetuar o estudo experimental dos vários algoritmos:

1. Algoritmo de pesquisa local (trepa-colinas, recristalização simulada ou outro, desde que aprovado pelo professor que lhe vai fazer a defesa)
 - Proponha uma representação que evite soluções inválidas;
 - Devem ser exploradas pelo menos duas vizinhanças diferentes.

2. Algoritmo evolutivo
 - Use a mesma representação que utilizou no algoritmo de pesquisa local;
 - Devem ser explorados pelo menos dois operadores de recombinação e dois operadores de mutação diferentes;
 - Devem ser explorados dois métodos de seleção diferentes.
3. Método híbrido combinando as duas abordagens anteriores
 - Devem ser exploradas pelo menos duas abordagens híbridas diferentes usando os dois algoritmos implementados nos pontos 1 e 2.

4. Estudo experimental

O estudo experimental deve analisar os parâmetros e componentes de cada algoritmo que podem ter influência no seu desempenho.

No algoritmo de pesquisa local deve realizar o estudo experimental variando:

- Número de iterações
- Se optar pela recristalização simulada: valores de T_{min} , T_{max} , $f_{arrefecimento}$
- Vizinhanças
- Aceitar soluções de custo igual

No algoritmo evolutivo deve realizar o estudo experimental variando:

- Tamanho da população
- Operadores de recombinação / mutação
- Probabilidades do operador de recombinação / mutação
- Métodos de seleção

Abordagens híbridas

- Escolher os melhores algoritmos encontrados no estudo experimental anterior para construir as abordagens híbridas (melhor pesquisa local + melhor evolutivo)
- Comparar as duas abordagens híbridas
- Apresentar uma tabela de comparação com os resultados obtidos por:
 - Melhor pesquisa local
 - Melhor evolutivo
 - Abordagem híbrida 1
 - Abordagem híbrida 2

As experiências devem ser repetidas pelo menos 10 vezes e as conclusões do estudo devem ser baseadas na comparação dos valores médios.

5. Critérios de avaliação

- Originalidade e correção dos algoritmos implementados:
 - Pesquisa local (10%)
 - Algoritmo evolutivo (20%)
 - Abordagens híbridas (10%)

- Experimentação e análise:
 - Pesquisa local (15%)
 - Algoritmo evolutivo
 - Impacto de diferentes operadores (15%).
 - Impacto da variação dos parâmetros (20%).
- Documentação e defesa (10%)

6. Normas de realização do trabalho prático

- O trabalho deve ser realizado **em grupos de dois alunos**.
- Os trabalhos serão sujeitos a defesa obrigatória, em data a anunciar.
- Cotação do trabalho: **6 valores**

7. Entrega do Trabalho

- Data limite para entrega: **23h59m do dia 09 de janeiro de 2023**
- Será dada uma penalização de **25%** por cada dia de atraso;
- Deverá ser submetido um ficheiro compactado em formato ZIP, devidamente identificado com os nomes e números dos alunos que realizaram o trabalho. Esse ficheiro deverá conter o seguinte:
 - **Relatório** abordando, pelo menos, os seguintes pontos:
 - Representação usada para o problema, descrição da função de avaliação e objetivo da otimização;
 - Descrição dos algoritmos e/ou das heurísticas utilizadas. Explicar quais as vizinhanças, métodos de seleção e operadores genéticos implementados;
 - Justificação das principais opções tomadas;
 - Resultados dos testes efetuados e respetiva análise. Os resultados a mostrar no relatório devem ser apenas um resumo baseado em tabelas/gráficos apresentando médias de várias repetições e respetivas conclusões. O estudo completo deve ser anexado num ficheiro Excel.
 - **Código fonte** do programa, executável e exemplos para teste;
 - **Ficheiro Excel** com os testes realizados para cada algoritmo.
- O trabalho deve ser entregue através da plataforma *Moodle* até à data limite indicada.
- As defesas serão nas aulas práticas da semana 09 a 13 de janeiro e noutros dias a combinar caso seja necessário. As defesas são obrigatórias e serão agendadas com os professores das aulas práticas.