

# Pesquisa aplicada à determinação de trajetos em viagens

Relatório final



Universidade do Porto

Faculdade de Engenharia

**FEUP**

Inteligência Artificial

3º ano

Mestrado Integrado em Engenharia Informática e Computação

Turma 5 - Grupo A1\_2

José Peixoto	200603103	ei12134@fe.up.pt
Pedro Moura	201306843	up201306843@fe.up.pt

29 de Maio de 2016

# Conteúdo

<b>1</b>	<b>Objetivo</b>	<b>2</b>
<b>2</b>	<b>Especificação</b>	<b>2</b>
2.1	Principais conceitos . . . . .	2
2.2	Descrição do problema . . . . .	3
2.3	Restrições do problema . . . . .	3
2.4	Representação do problema . . . . .	3
2.5	Solução do problema . . . . .	4
2.6	Função do custo . . . . .	4
2.7	Funções heurísticas . . . . .	4
2.8	Explicação de datasets . . . . .	5
<b>3</b>	<b>Desenvolvimento</b>	<b>5</b>
3.1	Linguagens de programação, ferramentas e APIs . . . . .	5
3.2	Estrutura da aplicação . . . . .	5
3.3	Detalhes relevantes da implementação . . . . .	8
<b>4</b>	<b>Experiências</b>	<b>8</b>
<b>5</b>	<b>Conclusões</b>	<b>8</b>
<b>6</b>	<b>Melhoramentos</b>	<b>9</b>
<b>7</b>	<b>Recursos</b>	<b>9</b>
7.1	Bibliografia . . . . .	9
7.2	Software . . . . .	9
7.3	Elementos do grupo . . . . .	9
<b>8</b>	<b>Apêndice</b>	<b>10</b>
8.1	Manual do utilizador . . . . .	10

## Resumo

No âmbito da unidade curricular de Inteligência Artificial, foi-nos proposta a elaboração de um relatório final que documentasse todo o trabalho realizado no projeto da pesquisa aplicada à determinação de trajetos em viagens.

## 1 Objetivo

No tema da “Pesquisa aplicada à determinação de trajetos em viagens” pretende-se determinar o melhor percurso a realizar de automóvel, numa viagem de grande distância. Pressupõe-se uma aplicação de algoritmos de pesquisa informada, estudados nas aulas, que permitam encontrar a solução ótima deste problema.

## 2 Especificação

### 2.1 Principais conceitos

Neste problema, os principais conceitos são o automóvel, o local de partida, o local de destino, estações de serviço, hotéis, pontos com interesse turístico e o trajeto ou estrada que faz a ligação entre os locais.

**Automóvel** é o meio de transporte usado pela família do Sr. X e permitirá a deslocação entre o local de partida e o local de destino, através de estradas, podendo passar por estações serviço, pontos com interesse para visita e hotéis.

**Local de partida** é a localização física onde a família do Sr. X iniciará a sua viagem de férias.

**Local de destino** é a localização final aonde a família do Sr. X pretende chegar na sua viagem de férias.

**Estações de serviço** são locais que permitem o abastecimento de combustível ao automóvel.

**Hotéis** são locais que permitem a pernoita de toda a família do Sr. X.

**Pontos com interesse turístico** são os locais com interesse turístico, e de possível passagem obrigatória pela família do Sr. X durante as férias.

**Estradas** são todas as vias de comunicação que permitem que o automóvel se desloque entre os pontos atrás referidos. No caso de ser uma autoestrada, considera-se que a via tem custos associados.

## 2.2 Descrição do problema

Com os conceitos atrás descritos, torna-se simples a compreensão de uma descrição geral do problema: Um automóvel pretende deslocar-se de um ponto A para um ponto B, sendo necessário descobrir o caminho mais curto e fisicamente possível (ver sub-secção seguinte), minimizando os custos associados.

## 2.3 Restrições do problema

Para além de determinar a melhor solução, ou seja, o melhor trajeto a realizar pelo automóvel, é necessário ter em conta certas restrições que alteram a busca da solução e simulam, de forma mais realista, o problema:

**Tempo máximo de condução:** a distância pode não ser percorrível em 24 horas e muito menos nas 8 horas que se definiram como tempo máximo de viagem por dia.

**Controlo da quantidade de combustível:** a quantidade de combustível tem de ser mantida por forma a permitir o movimento do automóvel sem que este se esgote antes de alcançar uma estação de serviço ou o ponto de destino, caso já tenha percorrido todos os pontos com interesse turístico.

**Passagem pelos pontos com interesse turístico:** outra restrição na escolha do itinerário final é a passagem obrigatória pelos pontos com interesse turístico.

**Custos associados:** minimização dos custos necessários no itinerário solução. Preferência por estradas sem custos adicionais, e minimização de pernoitas, ou utilização de hotéis mais baratos.

## 2.4 Representação do problema

Para representar o mapa, é usado um grafo não dirigido com uma lista de arestas adjacentes em cada nó. Para simplificação na criação de dados de entrada, cada nó representa uma cidade que pode ou não ser um local obrigatório de passagem e contém estações de serviço e hotéis. As coordenadas geográficas de cada nó são representadas em graus decimais. As arestas do grafo guardam a distância, o tempo médio necessário para o percorrer e o seu preço ( $0 \text{ €}$  quando sem custos e  $> 0 \text{ €}$ , caso se trate de uma via de comunicação com custos de passagem).

Na representação do carro, para além da informação relativa à capacidade do tanque de combustível e autonomia média, guarda-se o plano de viagem pretendido, com os nós de passagem obrigatória, sob a forma de uma lista ligada em que o primeiro elemento é o ponto de partida, e o último o ponto de chegada.

## 2.5 Solução do problema

Na determinação da melhor solução para o problema, é aplicado o algoritmo  $A^*$  a um grafo que representa um mapa com alguns dos itinerários possíveis desde o local de partida até ao local de destino. Este algoritmo usa um método de “o melhor primeiro”, fazendo uma avaliação  $f(n)$  dos nós, combinando o custo necessário para se atingir o nó  $g(n)$  e o custo necessário para se alcançar o objetivo desse nó  $h(n)$ :

$$f(n) = g(n) + h(n) \quad (1)$$

## 2.6 Função do custo

No cálculo da função  $g(n)$ , é contabilizada a distância em quilómetros ( $km$ ) já percorridos até ao nó em questão e o dinheiro gasto em euros (€) em combustível, alojamento e portagens. A combinação de estes dois custos, quilómetros e euros, são feitos com uma combinação linear, determinada por dois coeficientes definidos pelo utilizador: `distanceRatio` e `priceRatio`.

## 2.7 Funções heurísticas

Para garantir a admissibilidade da função heurística  $h(n)$ , calcula-se a distância em linha reta entre as coordenadas geográficas do nó em questão e o local de destino, apenas no caso em que todos os locais com interesse turístico ( $POI$ ) já tenham sido visitados. A existirem  $POI$  com passagem obrigatória antes de chegar ao destino, é usada a divisão em subproblemas com atribuição dos  $POI$  como locais de destino intermédios, garantindo que se usa o processo atrás descrito. Na heurística avançada, a distância em linha reta entre as coordenadas geográficas é multiplicada por um fator (1.25), permitindo na maioria dos casos, uma menor geração de estados intermédios, perdendo no entanto, as qualidades de consistência e de admissibilidade.

$$h_1(n) = (distanceToGoal * distanceRatio) + (neededLitersToGoal * priceRatio) \quad (2)$$

$$h_2(n) = (1.25 * distanceToGoal * distRatio) + (neededLitersToGoal * priceRatio) \quad (3)$$

Nas duas funções heurísticas não são feitas estimativas acerca dos gastos futuros em € em hotéis nem em portagens, uma vez que tal não é possível sem comprometer, de forma significativa, a admissibilidade das heurísticas.

## 2.8 Explicação de datasets

Para guardar e ordenar os estados gerados, utiliza-se uma fila de prioridades, permitindo um rápido acesso ao primeiro elemento da fila (mais promissor). Os estados gerados e guardados na lista fechada, têm um acesso de tempo constante, uma vez que se armazenam num hashmap.

## 3 Desenvolvimento

### 3.1 Linguagens de programação, ferramentas e APIs

Quer o algoritmo quer a interface gráfica foram programadas em Java com auxílio do programa de ambiente de desenvolvimento **Eclipse**. Na interface gráfica usa-se o Java Swing nas janelas que permitem a configuração do itinerário e das variáveis do carro. Na visualização do mapa e da solução encontrada, recorre-se à biblioteca dinâmica de modelação e análise de grafos **GraphStream**.

### 3.2 Estrutura da aplicação

O projeto está dividido em três pastas: código fonte (src), documentos (doc) e recursos (res). Por forma a organizar o projeto, os ficheiros com o código fonte estão subdivididos em seis pacotes:

**agents** implementação do principal agente do problema: o automóvel.

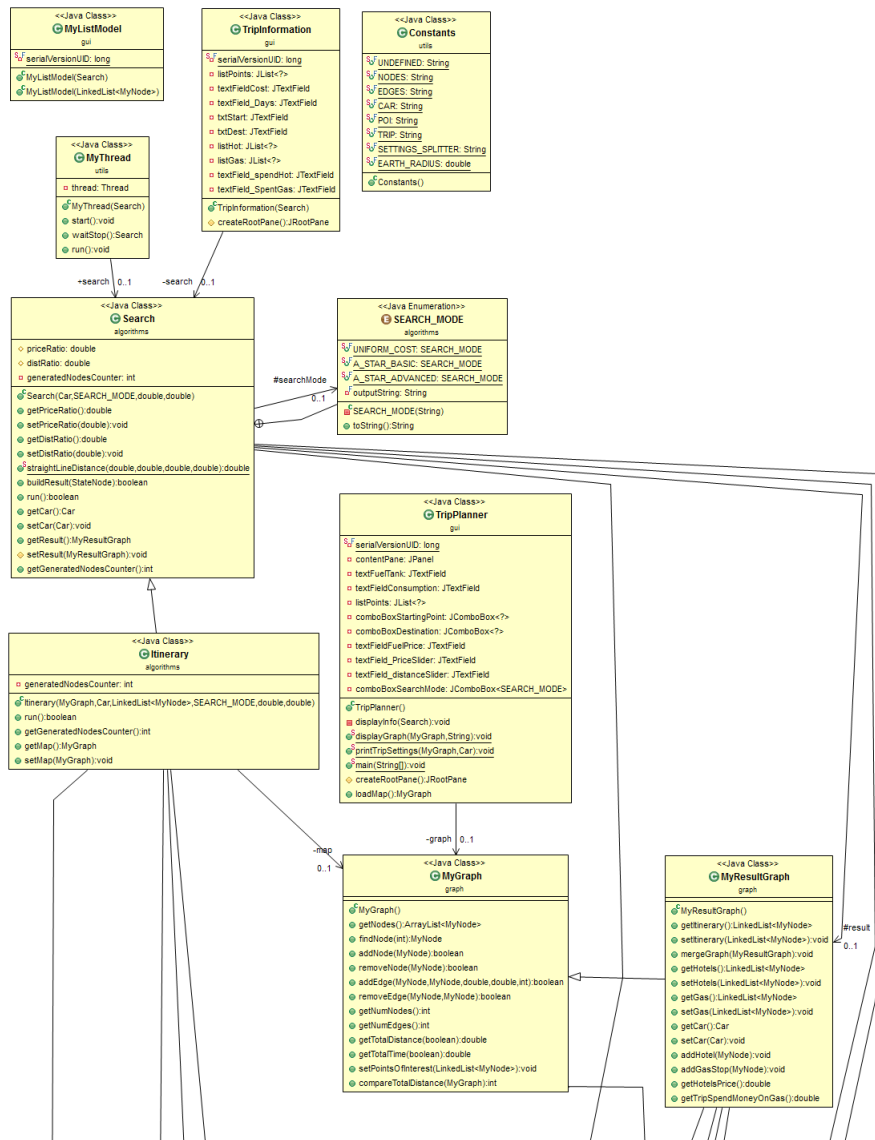
**algorithms** lógica da pesquisa da solução óptima do problema.

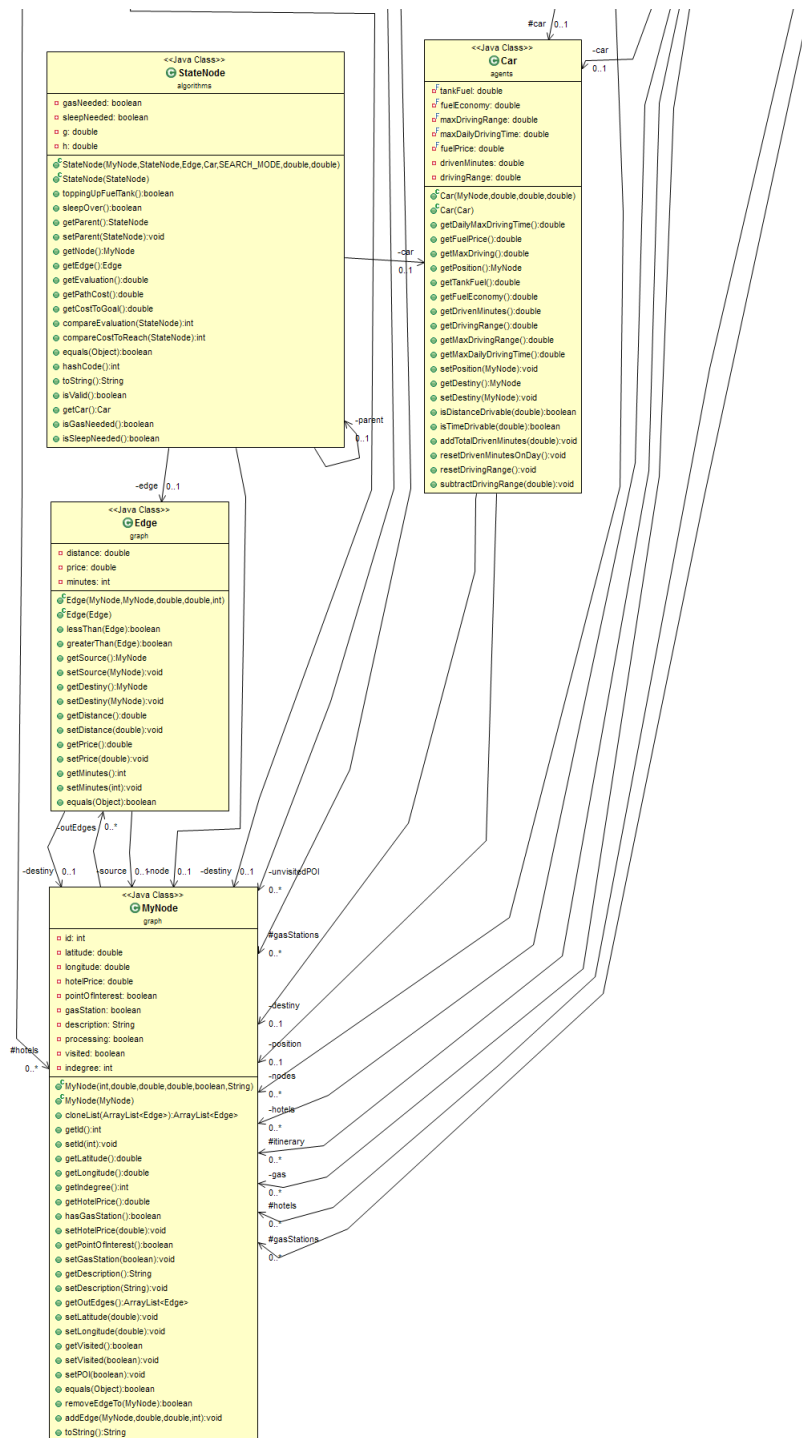
**graph** representação do mapa do problema com estradas e cidades.

**gui** interface gráfica que permite a interação simplificada do programa pelo utilizador.

**junit** ficheiros que permitem efectuar testes unitários a funções importantes do programa.

**utils** funções com utilidade na execução do programa mas que não se enquadram nos pacotes atrás descritos.







### 3.3 Detalhes relevantes da implementação

No caso de serem seleccionados pontos com interesse de visita (interpretados como de passagem obrigatória), são feitos vários cálculos de subproblemas nas pesquisas dos caminhos mais curtos. Para agilizar este processo, são criadas threads para cada uma destas pesquisas, permitindo o processamento em paralelo.

## 4 Experiências

Para além dos testes unitários que testam os componentes importantes do programa, foram também feitas experiências quanto à performance do algoritmo de pesquisa de soluções utilizado, comparando o método do custo uniforme (UCS), com as duas versões do algoritmo  $A^*$ . Em seguida, mostram-se os efeitos da precisão da função heurística na performance da busca de soluções, comparando o número de nós gerado e o valor da ramificação efectiva dada uma certa profundidade da solução:

Número de nós gerados				Factor efectivo de ramificação		
p	UCS	$A^*(h_1)$	$A^*(h_2)$	UCS	$A^*(h_1)$	$A^*(h_2)$
2	21	16	9	4.11	3.53	2.54
4	105	53	35	2.89	2.37	2.10
8	169	134	97	1.70	1.64	1.57
10	240	126	91	1.56	1.45	1.39
12	309	171	130	1.47	1.38	1.34
14	328	206	147	1.38	1.33	1.29
16	344	270	172	1.32	1.29	1.25
...	...	...	...	...	...	...

## 5 Conclusões

Após análise dos resultados, conseguimos concluir que o algoritmo do custo uniforme é bastante menos eficiente, uma vez que tem consistentemente um valor de ramificação efectiva superior às duas versões do algoritmo do  $A^*$ . Pela definição das duas funções heurísticas atrás referidas, conclui-se que para qualquer nó  $n$ ,  $h_2(n) \geq h_1(n)$ . Pode dizer-se que  $h_2$  domina o  $h_1$ . Nos testes acima apresentados, o domínio da função  $h_2$  também se traduziu em melhor performance.

## 6 Melhoramentos

Sugere-se que em trabalhos futuros se altere a forma de gerar/importar dos dados de entrada. Devido à difícil importação de dados com o formato e detalhes adequados ao problema, aconselha-se que seja feita uma geração automática de um grande número de nós e arestas, permitindo aumentar o tamanho do grafo que representa o mapa e consequentemente a complexidade dos problemas.

## 7 Recursos

### 7.1 Bibliografia

- Stuart Russell, Peter Norvig *Artificial Intelligence A Modern Approach*, Pearson Education 3rd edition, 2010.

### 7.2 Software

- Recolha de coordenadas: *Google Maps*, 2016
- Representação de grafos: <http://graphstream-project.org/> *GraphStream*, 2015

### 7.3 Elementos do grupo

Percentagem aproximada do trabalho efetivo de cada um dos elementos do grupo:

- José Peixoto - 50%
- Pedro Moura - 50%

## 8 Apêndice

### 8.1 Manual do utilizador

The screenshot shows the 'Trip Planner' application window. It is divided into three main sections: 'Car settings', 'Trip settings', and 'Points of interest'.  
**Car settings:** Includes input fields for 'Fuel tank capacity (L):' (50), 'Fuel consumption (L/100km):' (5), and 'Fuel price (€/L):' (1).  
**Trip settings:** Includes 'Starting point:' and 'Destination:' dropdown menus, both set to 'Porto'.  
**Points of interest:** A list box containing: Porto, Viseu, Coimbra, Lisbon (highlighted), Salamanca, Madrid (highlighted), Saragoza, Barcelona, Toulouse, Rodez, Faro, and Sevilha.  
**Solution search mode:** A dropdown menu set to 'A\* search with basic heuristic function'.  
**Relative cost coefficient ratio:** A slider bar.  
**Price (€):** Input field set to 0.0.  
**Distance (km):** Input field set to 1.0.  
**Buttons:** 'Show map', 'Plan trip', and 'Exit'.

Figura 1: Janela principal que permite ao utilizador a configuração das definições da viagem

The screenshot shows the 'Trip Information' application window, displaying the results of a trip plan. It is divided into three columns: 'Itinerary:', 'Hotels:', and 'Gas stations:'.  
**Itinerary:** A list of 8 stops: 1 - Porto, 2 - Coimbra, 3 - Lisbon, 4 - Merida, 5 - Madrid, 6 - Salamanca, 7 - Viseu, 8 - Porto.  
**Hotels:** A list with 1 - Merida.  
**Gas stations:** A list with 1 - Madrid.  
**Summary:** A table at the bottom showing trip statistics.

Start:	Spent on hotels (€):	Total distance driven (km):
Porto	90.0	1501.0
Destination:	Spent in fuel (€):	Time spent driving (hours):
Porto	75.05	17.966666666666665

Figura 2: Janela com a informação acerca da viagem gerada

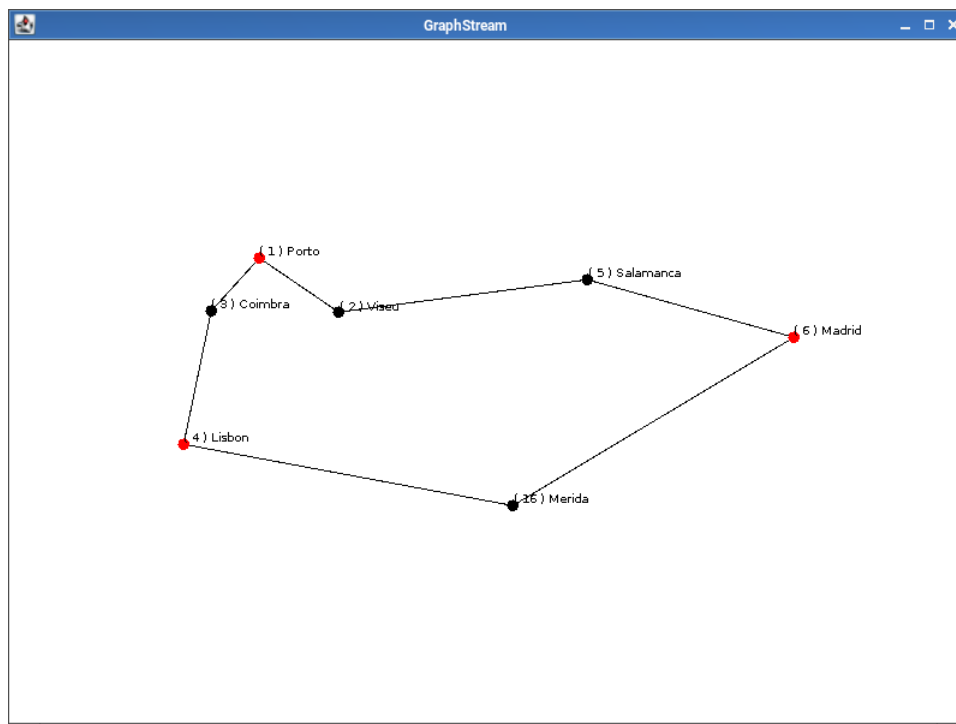


Figura 3: Janela que contém uma representação do mapa