

Universidade do Minho
Escola de Engenharia

Inteligência Artificial

Departamento de Informática

Trabalho em Grupo - Fase 2

Grupo 49

A93247 - André Lucas Silva Verdelho

A93184 - Diogo Camacho Barbosa

A93188 - Nelson Miranda Ribeiro

A89482 – Simão Brito

Índice

Introdução	3
Objetivos	4
Alterações da primeira fase	4
Arquitetura do Trabalho	5
Base de conhecimento	6
Remoção e Adição de Conhecimento	12
Predicados principais	13
Análise Crítica Global dos Resultados	14
Conclusão	16

Introdução

Este trabalho foi realizado para a cadeira de Inteligência Artificial e foi dividido em 2 fases. Na primeira fase foi-nos pedido para desenvolver um sistema de representação de conhecimento e raciocínio com capacidade para caraterizar um universo na área da logística de distribuição de encomendas. Para esta fase tivemos que declarar uma parte do conhecimento principal do nosso projeto, nomeadamente o conhecimento relacionado aos estafetas, entregas e veículos. De seguida, procedemos a implementar funcionalidades no sistema de modo a respeitar os requisitos do projeto – a estas funcionalidades chamamos de *queries* uma vez que maioritariamente serviam para nos informar de certas estatísticas em situações específicas.

Para a segunda fase do projeto foi-nos pedido para expandir o conhecimento do projeto, adicionando um grafo representativo de um cenário minimamente realista para locais de entrega. Para esta parte do projeto foi importante aplicar as estratégias diferentes para resolver os problemas em questão, utilizando algoritmos de procura para encontrar as soluções ótimas de caminhos de entrega das encomendas. Durante esta fase realizamos várias alterações que mais à frente vamos explicar no tópico dedicado às mudanças em relação à primeira fase.

Durante esta fase o, principal requisito era a simulação de entregas de encomendas, partindo de um estado inicial (o ponto de recolha de estafetas, ou seja, o centro de distribuição da *Green Distribution*) e realizando as entregas a determinados locais - pontos de entrega - e por fim retornando ao estado final (o centro de distribuição). Tivemos que implementar indicadores de produtividade dependendo do peso que cada tipo de transporte leva e também da distância percorrida. para finalizar, desenvolvemos uma maneira de cada estafeta poder passar a efetuar mais do que uma entrega em cada viagem que realiza.

Optamos por continuar a utilizar PROLOG, uma vez que chegámos à conclusão, após debater em grupo, que seria a linguagem mais adequada para este tipo de projeto.

Objetivos

O principal objetivo deste projeto era a simulação de um sistema de distribuição de entregas, tendo de recorrer a algoritmos de procura informados e não informados, com a finalidade de obter os melhores percursos para cada entrega a ser realizada. Como já referimos anteriormente, optámos por continuar a utilizar o PROLOG visando uma melhor aprendizagem de uma linguagem declarativa. para além de descobrir os melhores caminhos, também tínhamos como objetivo, descobrir qual o melhor algoritmo para cada situação, avaliando a sua performance para tal.

Alterações da primeira fase

Durante a segunda fase deste projeto, necessitamos de realizar algumas alterações ao conhecimento inicialmente definido, nomeadamente, para o conhecimento das encomendas, definimos que uma encomenda com 3 argumentos seria uma encomenda que já foi entregue, e para as encomendas com 4 argumentos, onde o 4º seria a morada, definimos que estas seriam as encomendas por fazer.

Mantivemos o raciocínio e a lógica das *queries* uma vez que estas não sofreram muitas consequências das alterações previamente mencionadas. Expandimos o nosso grafo inicialmente planeado e dividimos o projeto em ficheiros para simplificar a consulta do código.

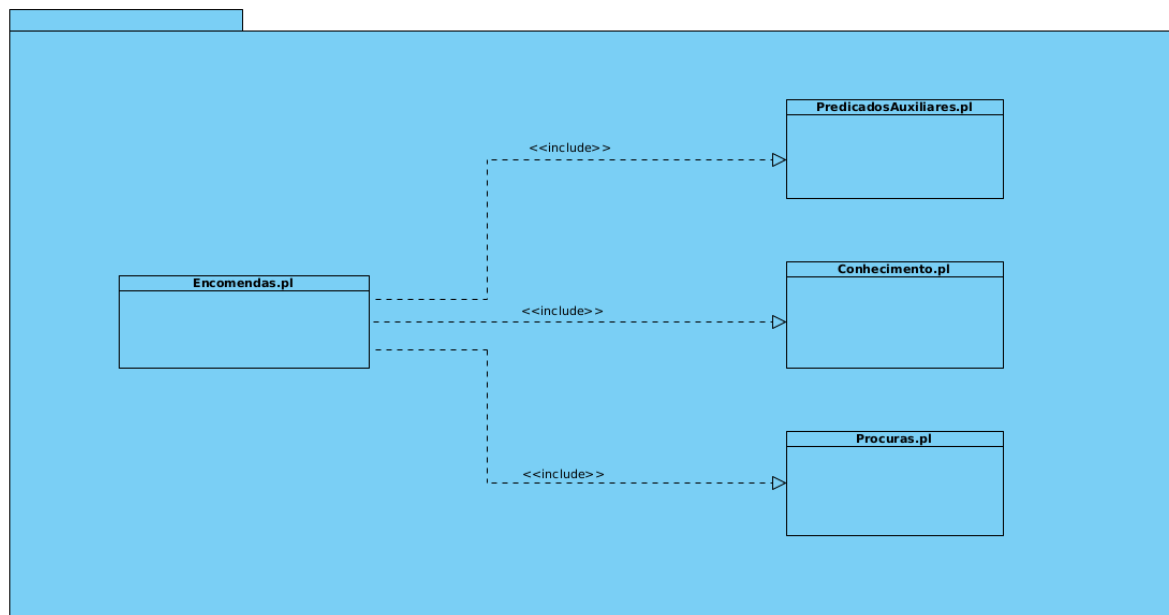
Arquitetura do Trabalho

O nosso projeto é composto por quatro ficheiros. O Encomendas.pl é o ficheiro principal que inclui os outros três ficheiros: Conhecimento.pl, Procura.pl e PredicadosAuxiliares.pl. Neste ficheiro estão implementados os predicados responsáveis por executar as principais funcionalidades do programa.

O ficheiro Conhecimento.pl, contém toda a base de conhecimento do projeto.

O ficheiro Procura.pl é onde estão definidos todos os algoritmos de procura que foram usados para percorrer o grafo em questão e para descobrir os caminhos entre nodos.

Por fim o ficheiro PredicadosAuxiliares.pl é onde se encontram todos os predicados que foram usados em auxílio à implementação das funcionalidades do programa.

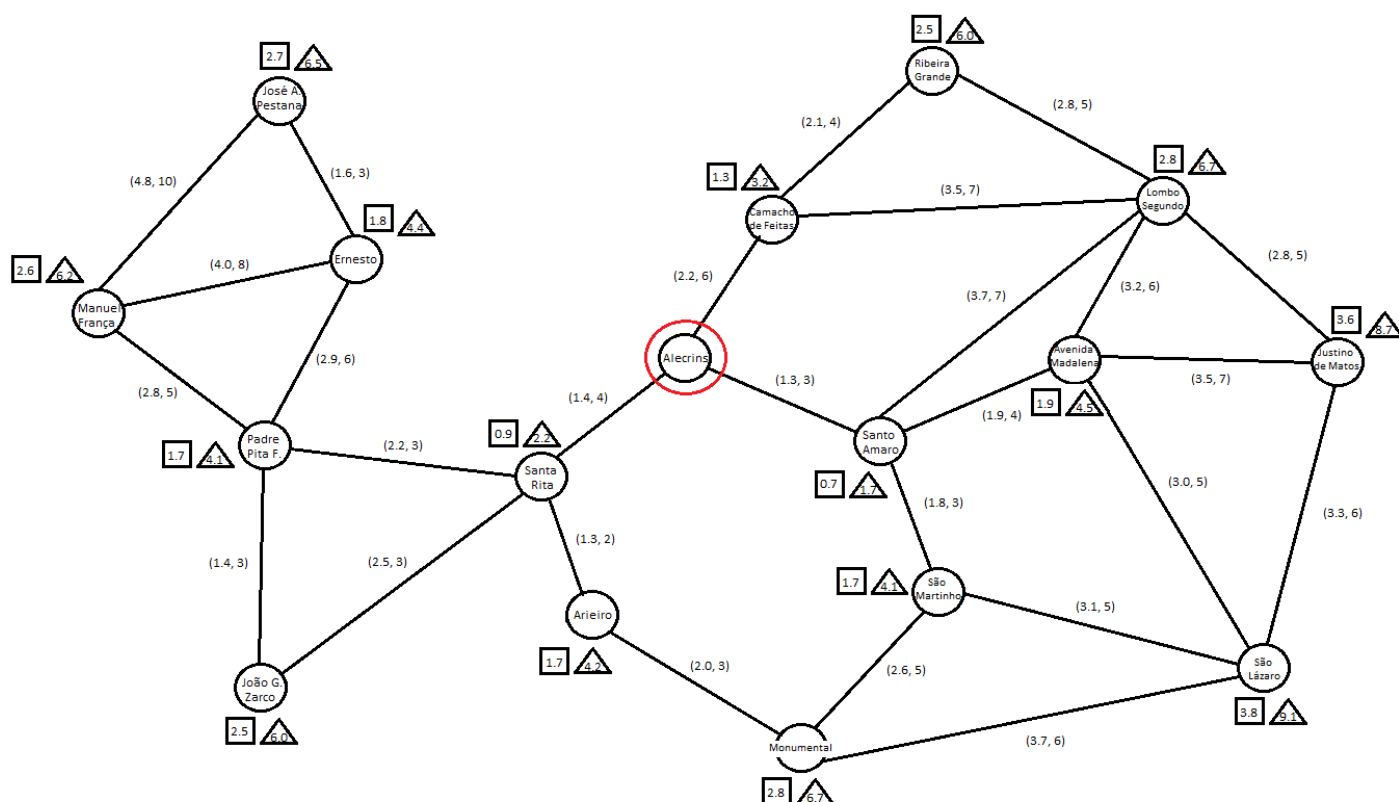


Base de conhecimento

O sistema de representação de conhecimento e raciocínio construído ao longo deste projeto, pretende retratar um universo de discurso na área de logística de distribuição de encomendas, por parte da empresa *Green Distribution*.

Uma vez que precisávamos de criar um cenário, minimamente realista, onde serão efetuadas as entregas de encomendas, escolhemos uma zona da parte sul da Madeira, mais nomeadamente no concelho de Funchal, criando assim um grafo que continha diversas ruas dessa região.

O grafo pode ser observável na figura seguinte:



Optamos por escolher como localização da empresa a Rua dos Alecrins, marcada com um círculo vermelho no grafo acima demonstrado.

```
localizacaoEmpresa("Rua dos Alecrins").
```

Os diversos pontos de entrega ficam apenas definidos pelas ruas existentes.

```
%pontoDeEntrega(Rua).  
%Pontos de entrega válidos.  
  
pontoDeEntrega("Rua de Santa Rita").  
pontoDeEntrega("Caminho de Santo Amaro").  
pontoDeEntrega("Estrada Comandante Camacho de Freitas").  
pontoDeEntrega("Rua do Arieiro").  
pontoDeEntrega("Estrada João Gonçalves Zarco").  
pontoDeEntrega("Rua Padre Pita Ferreira").  
pontoDeEntrega("Caminho do Ernesto").  
pontoDeEntrega("Rua Manuel França").  
pontoDeEntrega("Estrada José Ângelo Pestana de Barros").  
pontoDeEntrega("Estrada Monumental").  
pontoDeEntrega("Caminho de São Martinho").  
pontoDeEntrega("Rua da Ponte São Lázaro").  
pontoDeEntrega("Avenida da Madalena").  
pontoDeEntrega("Rua Álvaro Justino de Matos").  
pontoDeEntrega("Caminho do Lombo Segundo").  
pontoDeEntrega("Caminho da Ribeira Grande").
```

O duplo (X,Y), no grafo, representa os custos associados entre as diversas ruas nas arestas, onde X representa o custo da distância e Y o custo do tempo. As estimativas dos custos de cada nodo até ao nodo da Rua dos Alecrins, encontram-se representadas dentro do quadrado (estimativa da distância) e do triângulo (estimativa do tempo).

As arestas são caracterizadas por duas ruas que se conectam, pelo custo de distância e de tempo.

```
%aresta(Rua1,Rua2,CustoDistancia,CustoTempo).  
  
aresta("Rua dos Alecrins", "Rua de Santa Rita", 1.4, 4 ).  
aresta("Rua dos Alecrins", "Caminho de Santo Amaro", 1.3 , 3).  
aresta("Rua dos Alecrins", "Estrada Comandante Camacho de Freitas", 2.2 , 6).  
aresta("Rua de Santa Rita", "Rua do Arieiro", 1.3 , 2).  
aresta("Rua de Santa Rita", "Rua Padre Pita Ferreira", 2.2, 3).  
aresta("Rua de Santa Rita", "Estrada João Gonçalves Zarco", 2.5 , 3).  
aresta("Estrada João Gonçalves Zarco", "Rua Padre Pita Ferreira", 1.4, 3).  
aresta("Rua Padre Pita Ferreira", "Rua Manuel França", 2.8, 5).  
aresta("Caminho do Ernesto", "Rua Padre Pita Ferreira", 2.9, 6).  
aresta("Rua Manuel França", "Caminho do Ernesto", 4, 8).  
aresta("Estrada José Ângelo Pestana de Barros", "Caminho do Ernesto",1.6, 3).  
aresta("Rua do Arieiro", "Estrada Monumental",2, 3).  
aresta("Estrada Monumental","Caminho de São Martinho", 2.6, 5).  
aresta("Caminho de São Martinho", "Caminho de Santo Amaro", 1.8, 3).  
aresta("Rua da Ponte São Lázaro", "Caminho de São Martinho", 3.1, 5).  
aresta("Caminho de Santo Amaro", "Avenida da Madalena", 1.9, 4).  
aresta("Caminho do Lombo Segundo", "Estrada Comandante Camacho de Freitas", 3.5, 7).  
aresta("Caminho do Lombo Segundo", "Avenida da Madalena", 3.2, 6).  
aresta("Avenida da Madalena", "Rua Álvaro Justino de Matos", 3.5, 7).  
aresta("Caminho da Ribeira Grande", "Caminho do Lombo Segundo", 2.8, 5).  
aresta("Estrada Comandante Camacho de Freitas", "Caminho da Ribeira Grande", 2.1, 4).  
aresta("Rua Álvaro Justino de Matos", "Caminho do Lombo Segundo", 2.8, 5).  
aresta("Rua Álvaro Justino de Matos", "Rua da Ponte São Lázaro", 3.3, 6).  
aresta("Rua da Ponte São Lázaro", "Estrada Monumental", 3.7, 6).  
aresta("Caminho de Santo Amaro", "Caminho do Lombo Segundo", 3.7, 7).  
aresta("Avenida da Madalena","Rua da Ponte São Lázaro",3,5).  
aresta("Rua Manuel França","Estrada José Ângelo Pestana de Barros",4.8,10).
```

Uma vez que precisávamos que uma aresta fosse bidirecional, criamos o seguinte predicado, podendo então ser possível o deslocamento da rua A para a rua B e o seu inverso.

```
aresta(R1,R2,C,T) :- aresta(R2,R1,C,T),!.
```


As estimativas são caracterizadas por uma rua, a estimativa do custo da distância e a estimativa do custo de tempo, da rua em questão até à localização da empresa.

```
%estima(Rua,EstimativaCustoDistancia,EstimativaCustoTempo).  
  
estima("Rua dos Alecrins", 0, 0).  
estima("Rua de Santa Rita", 0.9, 2.2).  
estima("Caminho de Santo Amaro", 0.7, 1.7).  
estima("Estrada Comandante Camacho de Freitas", 1.3, 3.2).  
estima("Rua do Arieiro", 1.7, 4.2).  
estima("Estrada João Gonçalves Zarco", 2.5, 6.0).  
estima("Rua Padre Pita Ferreira", 1.7 , 4.1).  
estima("Caminho do Ernesto", 1.8, 4.4).  
estima("Rua Manuel França", 2.6, 6.2).  
estima("Estrada José Ângelo Pestana de Barros", 2.7, 6.5).  
estima("Estrada Monumental", 2.8, 6.7).  
estima("Caminho de São Martinho", 1.7, 4.1).  
estima("Rua da Ponte São Lázaro", 3.8, 9.1).  
estima("Avenida da Madalena", 1.9 , 4.5).  
estima("Rua Álvaro Justino de Matos", 3.6, 8.7).  
estima("Caminho do Lombo Segundo", 2.8, 6.7).  
estima("Caminho da Ribeira Grande", 2.5, 6.0).
```

Os transportes existentes utilizados pela empresa *Green Distribution* são: bicicletas, motos e carros. Estes são caracterizados pelo peso e volume máximo que conseguem transportar, pela velocidade média e por um nível de ecologia, sendo este um número mais elevado se se tratar de em veículo ecológico, como por exemplo, uma bicicleta.

```
%transporte(peso,velocidade media,volume,ecologia).  
  
transporte(bicicleta,5,10,20,10).  
transporte(mota,20,35,75,4).  
transporte(carro,100,25,300,1).
```

Os estafetas da empresa são apenas definidos pelo seu identificador e pelo seu nome.

```
%estafeta(ID,Nome).  
  
estafeta(1,"Gervásio Motoboy").  
estafeta(2,"Mario Visente").  
estafeta(3,"Mariano Díaz").  
estafeta(4,"Amélio Cardoso").  
estafeta(5,"Capelo Rego").
```

Os clientes são caracterizados pelo seu nome, identificador e por uma lista de triplos, contendo informação relativa à encomenda que recebeu, sendo este triplo composto: pelo identificador do estafeta que realizou a entrega, pelo identificador da encomenda e pelo *rating* que o cliente atribui à entrega realizada pelo estafeta.

```
%cliente(Nome,ID,[(EstafetaID/EncomendaID/Rating)]).  
  
cliente("Tomas Pina",1,[(4/1/2),(1/5/4)]).  
cliente("Graciano Nunes",2,[(1,8,5)]).  
cliente("Lídio Pereira",3,[(4/7/3),(5/9/5)]).  
cliente("Tomas Pontes",4,[(2/2/2),(1/4/5)]).  
cliente("Romario Faria",5,[(5/3/4),(5/6/4)]).  
cliente("Faustino Silva",6,[]).  
cliente("Ramiro Calcifo",7,[]).  
cliente("Jorge Fluído",8,[]).  
cliente("Cândido Faísca",9,[]).  
cliente("Teresa Pinto",10,[]).  
cliente("Maria Ribas",11,[]).  
cliente("Maria Joana",12,[]).  
cliente("Ana Cacho",13,[]).  
cliente("Susana Antunes",14,[]).  
cliente("Carolina Lima",15,[]).  
cliente("Rafaela Ferreira",16,[]).
```

As entregas realizadas e concluídas ficam caracterizadas pela: data de entrega, rua, identificador da encomenda, transporte, identificador do estafeta e identificador do cliente, ao qual foi entregue a encomenda.

```
%entrega(DataDeEntrega,Rua,EconmendaID,Transporte,EstafetaID,IDCliente).  
  
entrega(27/11/2021,"Rua Manuel França",1,mota,4,1).  
entrega(27/11/2021,"Estrada Comandante Camacho de Freitas",2,carro,4,4).  
entrega(11/11/2021,"Rua de Santa Rita",3,bicicleta,5,5).  
entrega(2/11/2021,"Estrada Comandante Camacho de Freitas",4,carro,1,4).  
entrega(27/10/2021,"Rua do Arieiro",5,carro,1,1).  
entrega(31/10/2021,"Caminho do Ernesto",6,bicicleta,5,5).  
entrega(6/10/2021,"Estrada José Ângelo Pestana de Barros",7,carro,4,3).  
entrega(15/10/2021,"Estrada Comandante Camacho de Freitas",8,mota,1,2).  
entrega(10/10/2021,"Caminho da Ribeira Grande",9,bicicleta,5,3).
```

As encomendas entregues são caracterizadas pelo peso, volume e pelo seu identificador.

```
%Encomendas entregues
%encomenda(Peso,Volume,EconmendaID).

encomenda(10,20,1).
encomenda(42,10,2).
encomenda(2,1,3).
encomenda(100,40,4).
encomenda(220,100,5).
encomenda(3,4,6).
encomenda(20,40,7).
encomenda(15,20,8).
encomenda(4,2,9).
```

As encomendas por entregar ficam também caracterizadas pelo peso, volume e identificador. Adicionamos agora a informação relativa à entrega, como por exemplo, a rua e o cliente.

```
%Encomendas por entregar
%encomenda(Peso,Volume,EconmendaID,Rua,IDCliente).

encomenda(10,10,10,"Rua Manuel França",6).
encomenda(4,9,11,"Rua de Santa Rita",7).
encomenda(15,30,12,"Estrada Comandante Camacho de Freitas",8).
encomenda(80,250,13,"Rua do Arieiro",9).
encomenda(1,2,14,"Estrada João Gonçalves Zarco",10).
encomenda(5,5,15,"Rua Padre Pita Ferreira",11).
encomenda(40,120,16,"Caminho do Ernesto",12).
encomenda(2,4,17,"Estrada José Ângelo Pestana de Barros",13).
encomenda(15,50,18,"Estrada Monumental",14).
encomenda(60,200,19,"Caminho de São Martinho",15).
encomenda(1,12,20,"Avenida da Madalena",16).
encomenda(10,70,21,"Rua Álvaro Justino de Matos",9).
encomenda(30,10,22,"Caminho do Lombo Segundo",11).
encomenda(5,5,23,"Caminho da Ribeira Grande",7).
encomenda(18,42,24,"Caminho de São Martinho",6).
encomenda(3,3,25,"Estrada João Gonçalves Zarco",13).
encomenda(130,47,26,"Estrada Comandante Camacho de Freitas",16). %pesoImp
encomenda(10,500,27,"Rua da Ponte São Lázaro",16). %volumeImp
encomenda(5,20,28,"Caminho do Lombo Segundo",10). %bikeE
encomenda(20,75,29,"Estrada João Gonçalves Zarco",11). %motaE
encomenda(100,300,30,"Rua da Ponte São Lázaro",14). %carroE
```

Remoção e Adição de Conhecimento

A adição e remoção de conhecimento ocorre sempre que uma encomenda é entregue.

Começamos por adicionar a encomenda/3 (encomenda entregue) ao conhecimento e calcular a data atual para podermos registar a entrega na base de conhecimento. Seguidamente é registada a encomenda no cliente que a vai receber e por fim, é removida a encomenda/5 (encomenda por entregar) da base de conhecimento.

```
%registo da entrega na base de conhecimento
registarEntrega(Peso,Volume,ID,Morada,T,IDEstafeta,ClienteID) :-

    %adição da encomenda concluída ao conhecimento
    assert(encomenda(Peso,Volume,ID)),

    %cálculo da data de entrega e adição da mesma na base de conhecimento
    get_time(X),
    stamp_date_time(X,date(Ano,Mes,Dia,_,_,_,_,_),'UTC'),
    assert(entrega(Dia/Mes/Ano,Morada,ID,T,IDEstafeta,ClienteID)),

    %adição da encomenda à lista de entregas do cliente
    random(1,6,Rating),
    cliente(Nome,ClienteID,Encomendas),
    retract(cliente(Nome,ClienteID,Encomendas)),
    append(Encomendas,[IDEstafeta/ID/Rating],NovasEncomendas),
    assert(cliente(Nome,ClienteID,NovasEncomendas)),

    retract(encomenda(Peso,Volume,ID,Morada,ClienteID)).
```

Predicados principais

entregaUmaEncomenda

Para efetuar a entrega de uma única encomenda é primeiramente selecionada a encomenda a entregar e o algoritmo de procura (informada) que será usado para fazer o cálculo do circuito a percorrer.

De seguida, é escolhido o estafeta e é verificado se o ponto de entrega é válido, ou seja, se existe na nossa base de conhecimento e é feita a escolha do transporte a utilizar, tentando sempre optar pela opção mais ecológica, sendo apenas limitado pelo peso e volume da encomenda.

Por fim o algoritmo de procura passado por input determina o circuito (ida e volta) que o estafeta tem de fazer para entregar a encomenda ao cliente tal como a distância a ser percorrida e o tempo que será necessário para completar a entrega, tendo em conta o veículo selecionado.

entregaVariasEncomendas

Para efetuar a entrega de várias encomendas é selecionada a lista de encomendas a entregar e os algoritmos de procura (informada e não informada) a serem utilizados no cálculo do circuito do estafeta.

Em primeiro lugar é calculado o peso e volume total das encomendas a serem entregues neste circuito para poder escolher um veículo mais ecológico que seja capaz de carregar a todas de uma vez, tal como o impacto que a carga a ser transportada terá no tempo necessário para completar todas as entregas.

De seguida, é calculado o caminho da base até ao local da primeira entrega com um algoritmo de procura informada, a partir dessa entrega é usado um algoritmo de procura não informada para calcular o caminho que o estafeta terá de percorrer para chegar ao próximo ponto de entrega.

Por fim, depois da última entrega, é usado de novo um algoritmo de procura informada para voltar à base. Em cada entrega feita, é registada no conhecimento e o peso total do veículo é decrementado para que o cálculo do tempo que o circuito demorará a ser concluído seja mais preciso.

Análise Crítica Global dos Resultados

Durante a implementação do código tentámos sempre recorrer aos melhores métodos de programação dos quais tínhamos conhecimento, contudo, sendo uma linguagem com a qual nunca tínhamos tido nenhum tipo de contacto, acrescido ao facto de ser uma linguagem declarativa, deparámo-nos com vários obstáculos e impedimentos ao rápido desenvolvimento do projeto.

Apesar das dificuldades que enfrentávamos com cada nova função que precisávamos de implementar, sentimos que estamos aptos para fazer uma análise crítica aos resultados que obtivemos, assim como também conseguimos fazer uma avaliação da performance de cada um dos algoritmos de procura em grafos utilizados.

Algoritmo	Tempo (ms)	Complexidade Temporal	Ótima
DFS	8	$O(b^m)$	Não
BFS	7	$O(b^d)$	Para custos escalonados iguais
Gulosa	4	Pior Caso : $O(b^m)$ Melhor Caso : $O(bd)$	Não
A*	5	-----	Sim

- m: profundidade máxima da árvore;
- d: profundidade da solução;
- b: máximo fator de ramificação da árvore de pesquisa;

Após fazer uma análise da teoria destes algoritmos, concluímos que o mais adequado para o modelo que optamos por desenvolver seria o algoritmo de procura A*. Este algoritmo é capaz de nos devolver a resposta ótima, e o seu desempenho é suficientemente rápido e, uma vez que dispomos de toda a informação necessária para o utilizar, selecionamos este para efetuar a procura dos caminhos para as entregas por fazer.

A procura em A* tem uma complexidade temporal variável, contudo tem uma complexidade espacial de $O(b^d)$, uma vez que armazena todos os nodos em memória, deste modo, em planeamento de rotas, é geralmente inferior à procura Gulosa, contudo, como já comprovamos, a procura A* é a melhor escolha em vários casos.

Observamos também que os algoritmos de Depth e Breadth-First Search demonstraram ter o pior desempenho visto que eram buscas não informadas, e apesar de o DFS ter uma possibilidade de não devolver o caminho ótimo, concluímos também que estes algoritmos seriam excelentes para travessias genéricas de grafos e/ou quando não possuímos a heurística para utilizar a busca A*.

Notamos que durante a testagem dos algoritmos a pesquisa Gulosa descobria sempre o caminho ótimo. Isto poderá significar que a distância real e a distância estimada entre os nós do nosso grafo coincidem.

Finalizando, compreendemos que os algoritmos de busca estudados e testados são essenciais para descobrir o caminho mais eficiente possível, contudo, a eficiência depende dos parâmetros que pretendemos otimizar. Neste caso, optamos por escolher os caminhos conforme a menor distância que irá ter de ser percorrida, levando a um serviço, geralmente, mais rápido e, conseqüentemente, mais ecológico e com maior volume de entregas num espaço de tempo.

Conclusão

Como grupo, consideramos que o projeto foi um excelente desafio e possibilitou uma consolidação dos conceitos lecionados nas aulas teóricas e, recorrendo a esta aprendizagem prática, facilitou a compreensão dos temas abordados. Acreditamos que o projeto foi essencial para o nosso grupo, contudo não estamos completamente satisfeitos com o resultado final do projeto.

Inicialmente concluímos que PROLOG seria a linguagem mais adequada a este projeto, contudo, consideramos que fomos um pouco limitados pela nossa ausência de conhecimento e prática com esta língua, pelo que não desenvolvemos o projeto com todos os requisitos a funcionar e corretamente implementados, o que nos levou, numa fase mais final, a ficar cada vez mais insatisfeitos com a ausência de funcionalidades extra.

Por outro lado, a utilização desta linguagem revelou ser algo inovador que quebrou o molde ao qual estávamos habituados de paradigmas funcionais e orientados a objetos, e, deste modo, forçou uma evolução das nossas capacidades de programação, levando-nos a aceitar que foi a melhor escolha que poderíamos ter feito.

Em suma, consideramos que o projeto foi um sucesso no âmbito didático, contudo não estamos completamente satisfeitos com o resultado final e cremos que existe um potencial para fazer um projeto melhor.