

**T02**

**Tráfego numa Cidade**

*Relatório Intercalar*

Agentes e Inteligência Artificial Distribuída

4º ano do Mestrado Integrado em Engenharia Informática e Computação

2017/2018

Elementos do Grupo:

Andreia Rodrigues – up201404691 – up201404691@fe.up.pt

Eduardo Leite – gei12068 – gei12068@fe.up.pt

Francisco Queirós – up201404326 – up201404326 @fe.up.pt

05/11/2017

**Índice**

[Enunciado](#_7bffnx9obvt9)

[1.1 Descrição do cenário](#_efbv3g2la17b)

[1.2 Objectivos do trabalho](#_6yvy745ubp7r)

[1.3 Resultados esperados e forma de avaliação](#_44xzlkqoqsxe)

[Plataforma/Ferramenta](#_9w3p7fuxw0bs)

[2.1 Para que serve e Descrição das características principais](#_knae2wwi8ehp)

[2.1.1 JADE](#_uoytt4n7hihb)

[2.1.2 Repast Simphony](#_jd8104dgqe7i)

[2.1.3 SAJaS](#_dsjyemgwcpj2)

[2.2 Realce das funcionalidades relevantes para o trabalho](#_8c2mshj8jbju)

[Especificação](#_mviffsvvaepg)

[3.1 Identificação e caracterização dos agentes (arquitectura, comportamento, estratégias)](#_n3o46jb450y)

[3.1.1 Veículos](#_69nfon7jq35l)

[3.1.2 Semáforos](#_gtpj1ecwayvg)

[3.2 Protocolos de interacção](#_bdp52g90ntqh)

[3.3 Faseamento do projecto](#_qah2k21cafwk)

# 

# 

# Enunciado

## 1.1 Descrição do cenário

No âmbito da unidade curricular de Agentes e Inteligência Artificial Distribuída o grupo propôs-se a desenvolver um programa que simula o tráfego automóvel numa cidade.

Este programa vai simular a interação entre automóveis face ao trânsito que se pode fazer ou não sentir devido à abundância de carros e aos vários semáforos espalhados pelas ruas dessa cidade. Os vários automóveis devem ser capazes de comunicar entre si o estado do trânsito onde circulam e os semáforos devem ser capazes de comunicar o seu estado a todos os automóveis existentes. O objetivo é que os automóveis evitem zonas com maior trânsito quando lhes for possível, chegando mais rápido ao seu destino.

## 1.2 Objectivos do trabalho

Com este trabalho, espera-se conseguir desenvolver agentes que quando estejam a funcionar em conjunto, permitam simular um ambiente fiel ao que nos foi pedido desenvolver.

Espera-se também conseguir que os agentes que representam os condutores/automóveis naveguem de forma eficaz em função da comunicação feita entre os agentes e o seu destino.

## 1.3 Resultados esperados e forma de avaliação

Espera-se observar minimização de engarrafamentos exceto em zonas sem acessos alternativos (dados importante para a gestão de itinerários) e uma minimização do tempo de viagem de cada um dos agentes que representa um automóvel.

Vamos avaliar estes fatores a partir de estatísticas obtidas a partir de cada uma das simulações dos agentes.

# Plataforma/Ferramenta

## 2.1 Para que serve e Descrição das características principais

### 2.1.1 JADE

O JADE (*Java Agent Development Framework*) é uma framework que simplifica a implementação de sistemas multi-agente de acordo com as especificações da FIPA (*Foundation of Intelligent Physical Agents*). Este software tem um sistema multi-agente, distribuído os agentes por *containers*, restringindo os agentes a um determinado domínio. Cada agente tem *behaviours* atribuídos, tarefas que podem ser executadas concorrentemente ou não. O JADE permite ainda a mobilidade e clonagem de agentes.

### 2.1.2 Repast Simphony

O Repast (*Recursive Porous Agent Simulation Toolkit*) é usado para modelar sistemas de agentes e fazer simulações utilizando agentes. Não é uma ferramenta para desenvolver os próprios sistemas de agentes, ao contrário do JADE. Com esta ferramenta é possível fazer uma recolha de dados e mostrar a interação entre os agentes durante a simulação através de uma interface.

### 2.1.3 SAJaS

O SAJaS (*Simple API for JADE-based Simulations*) é uma API cujo objetivo é servir de ponte entre a simulação de sistemas multi-agente e desenvolvimento dos mesmos. Através do SAJaS é possível beneficiar de ambas as funcionalidades do JADE e do Repast.

## 2.2 Realce das funcionalidades relevantes para o trabalho

A partir da utilização da framework JADE vai ser possível a utilização das abstrações já existentes de *Agent* e *Behaviour*, assim como a comunicação *peer-to-peer* entre os diferentes agentes.

O Repast + SAJaS vai permitir simulações com os agentes criados em JADE, visualização das interações entre os mesmos e obtenção dos resultados em gráficos para posterior análise.

# Especificação

## 3.1 Identificação e caracterização dos agentes (arquitectura, comportamento, estratégias)

### 3.1.1 Veículos

Os veículos terão em cada instante uma posição e um local aonde querem chegar.

Para concluírem o seu objetivo vão evitar usar caminhos congestionados de forma a minimizar o tempo necessário. A estratégia a usar será:

1. Determinar o caminho ótimo para chegar ao seu destino a partir da sua localização atual usando caminhos pré-determinados ou usando métodos de *pathfinding* (A\*, por exemplo).
2. A cada ponto de decisão de direção (ou seja, cruzamentos ou entroncamentos), analisar o troço a seguir escolhido no caminho ótimo. Se este estiver impedido ou for muito prejudicial em termos de tempo, tentar escolher um caminho que não use este troço. Caso determine um caminho usando outro troço, dirige-se a esse troço e usa o novo caminho escolhido. Caso contrário, usa o troço original e mantém o caminho determinado no passo 1.

Estes agentes também irão parar caso agentes do tipo semáforo que estejam presentes no mesmo troço mandem parar.

### 3.1.2 Semáforos

Os semáforos são agentes que podem ter dois estados: verdes ou vermelhos. Estes agentes vão regularmente trocar o seu estado de forma a regularizar o trânsito.

Estes agente não terão estratégias complexas, apenas irão trocar o seu estado num intervalo fixo. Tem apenas o papel de condicionar os agentes veículo.

## 3.2 Protocolos de interacção

Usando as capacidades de comunicação entre agentes do JADE, os agentes irão comunicar entre si.

Os agentes semáforo comunicam aos agentes veículo sobre o seu estado. Em função deste estado, os agentes veículo param ou avançam no troço relevante. Para este efeitos, existe um simples protocolo de os semáforos mandarem ao veículos uma mensagem sobre o seu estado.

Os agentes veículo comunicam entre si para nos cruzamentos indicar o sentido que querem tomar, de forma a cederem a passagem segundo a regra geral de cedência de passagem do Código da Estrada de Portugal.

## 3.3 Faseamento do projecto

1. Numa fase inicial, vai-se utilizar mapas fixos e implementar o comportamento básico dos veículos (navegação). Para este efeito podem-se usar caminho predefinidos, mas mantendo, o mais possível, a lógica dentro dos agentes.
2. Introduzir os semáforos e condicionar os veículos à regularização dos semáforos.
3. Permitir uma comunicação entre veículos (mudança de direção, por exemplo).
4. Introduzir comunicação rádio que todos os veículos recebem simultaneamente (para indicar o fecho de uma rua, por exemplo).
5. Considerar mapas dinâmicos (usando grafos, por exemplo) e criar a capacidade de *pathfinding* nos agentes veículo. Isto pode mostrar-se difícil de executar e talvez não haja tempo para cumprir este ponto.

**Recursos**

**4.1 Bibliografia**

<https://paginas.fe.up.pt/~eol/AIAD/aulas/jade_en.pdf>

<https://en.wikipedia.org/wiki/Java_Agent_Development_Framework>

<http://jade.tilab.com/>

<https://en.wikipedia.org/wiki/Repast_(modeling_toolkit)>

<https://repast.github.io/>

<https://paginas.fe.up.pt/~eol/AIAD/aulas/REPAST_intro_hlc.pdf>

<https://web.fe.up.pt/~hlc/doku.php?id=sajas>

**4.2 Software**

<http://jade.tilab.com/>

<https://repast.github.io/repast_simphony.html>

<https://web.fe.up.pt/~hlc/doku.php?id=sajas>