



**FEUP** FACULDADE DE ENGENHARIA  
UNIVERSIDADE DO PORTO

Mestrado Integrado em Engenharia Informática e  
Computação

**Agentes e Inteligência Artificial Distribuída**

## *Coordenação inteligente de semáforos*

**Grupo L:**

João Guimarães Bernardo - 201005324 - mieic1005324@fe.up.pt

Liliana Lobo Ribeiro - 201200675 - ei12077@fe.up.pt

Pedro Lemos Faria - 201206054 - ei12097@fe.up.pt

Faculdade de Engenharia da Universidade do Porto  
Rua Roberto Frias, sn, 4200-465 Porto, Portugal

8 de Novembro de 2015

# 1 Enunciado

Este trabalho tem como objetivo explorar diferentes estratégias de coordenação de semáforos usando um Sistema Multi Agente, a partir da configuração inteligente dos parâmetros associados à temporização das luzes, utilizando a metáfora dos agentes inteligentes.

Através das informações fornecidas por sensores os semáforos têm acesso ao volume de tráfego atual. Estas informações serão usadas para o agente (semáforo) selecione o melhor plano/estratégia para controlar o seu comportamento.

Diferentes estratégias devem ser testadas, nomeadamente: a possibilidade de comunicação entre vários semáforos vizinhos, quer com o objetivo de enriquecer a circulação de informação entre os agentes, quer com vista a uma coordenação multi agente mais eficaz; a exploração de estratégias baseadas em mercado (ver artigo abaixo).

A análise da qualidade de cada estratégia poderá ser realizada através de medidas de desempenho tais como o tempo de espera nas filas, a velocidade média na rede, ou a razão entre veículos entrados e saídos. O trabalho inclui a construção de um modelo de simulação de tráfego ou a utilização do SUMO (Simulation of Urban MObility).

## 2 Plataforma/Ferramenta

Para a elaboração deste trabalho vamos usar a framework JADE ( Java Agent Development Framework), esta framework é implementada usando java, é na realidade uma API com vários packages. Uma das vantagens em usar JADE é que a implementação de sistemas de multi agentes é simplificada, uma vez que esta segue as especificações FIPA, possui uma interface gráfica que permite perceber melhor o que se está a passar entre os agentes, através de agentes chamados de sniffers. Isto simplifica muito a parte de debugging da aplicação que seria muito difícil de realizar usando os métodos mais tradicionais.

Os agentes podem ser distribuídos por diferentes plataformas/máquinas, sem ser necessário que essas máquinas partilhem o mesmo sistema operativo.

Para além de oferecer uma abstração na construção de um agente, JADE permite uma comunicação assíncrona entre agentes, um serviço de “paginas amarelas” ao qual os agentes podem subscrever tal como descobrir outros agentes que também estejam subscritos. Deste modo os agentes podem subscrever com um atributo caracterizador, como por exemplo a sua posição, ou o cruzamento a que pertencem o que irá facilitar a comunicação com outros agentes que podem simplesmente mandar mensagem para o semáforo que querem falar em vez de mandar mensagens para todos e esperar pela resposta do semáforo que na realidade precisavam de comunicar.

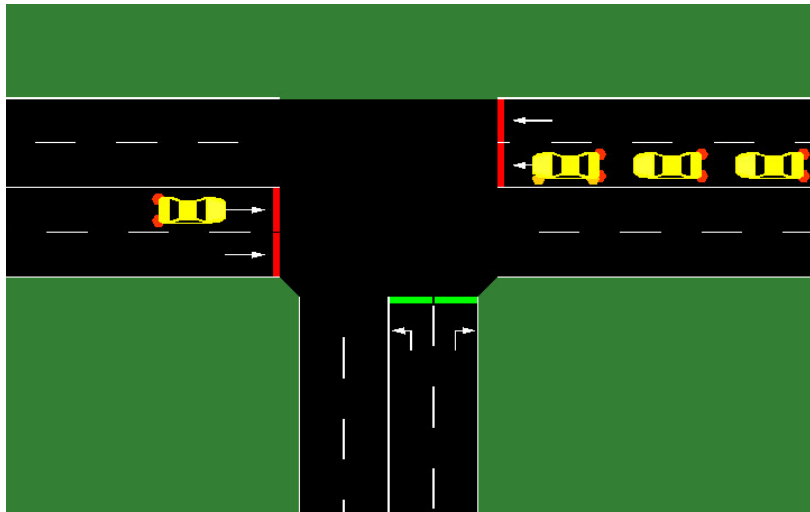
### 3 Especificação

O conjunto de semáforos presente num cruzamento ou intersecção é representado por um agente chamado TrafficLight, que tem como atributos a informação da cor atual de cada um dos semáforos desse conjunto e a sua localização no sistema.

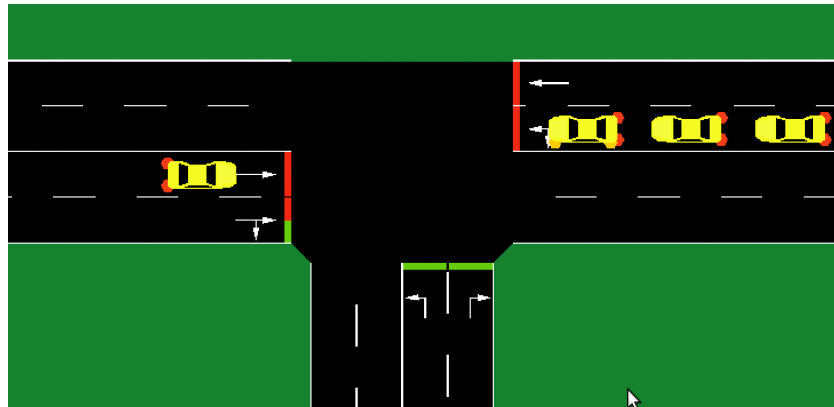
Um agente tem um conjunto de sensores que fornecem informações de tráfego em tempo real, assim quando uma rua do cruzamento tem excesso de tráfego um trigger é accionado que ira causar a troca das cores atuais de cada semáforo dentro do agente. Caso vários triggers sejam accionados ao mesmo tempo, a acção que corresponde ao trigger com mais prioridade sera selecionado. Num cenário em que nenhum trigger é accionado nem nenhuma mensagem for recebida pelo agente este regula as cores dos semáforos segundo um contador que varia com a altura do dia ( alturas de maior transito ou de menor, irão ter contadores com valores diferentes ).

A interação entre agentes é feita através de mensagens, mais precisamente mensagens que cumprem as especificações ACL.

Numa primeira fase irá ser desenvolvido o sistema tendo em conta um cenário muito básico em que um agente é constituído apenas por um semáforo, apenas para se implementar comunicação entre agentes, e garantir que o agente se consegue regular a si mesmo.



Numa fase posterior o sistema será estendido para funcionar com cenários mais complexos, em que o agente poderá ter mais do que um semáforo, e o conjunto dos seus semáforos não é independente entre si ( ou seja a mudança da se cor do semáforo implica a mudança de cor de outro semáforo do mesmo agente). Isto implica que a alteração de cores dos semáforo de um agente irá ter influencia no semáforos dos agentes que são adjacentes a este.



Numa fase final ira ser implementado um sistema de prioridades, em que se um agente for considerado prioritário quando “pede” a outros agentes para alterarem de cor esses agentes irão ignorar os triggers acionados pelos seus sensores e irão (se possível) obedecer a esse pedido. Nesta fase, pretendemos também implementar um percurso completo, permitindo a que um veiculo possa passar mais do que uma vez no mesmo semáforo ou conjunto de semáforos.

## 4 Recursos

### 4.1 Software

Software que irá ser usado para o desenvolvimento do sistema:

- Java IDE - Eclipse
- Simulação de trafego - SUMO
- JADE
- TraSMAPI

#### 4.1.1 Bibliografia

- <https://github.com/tjiagoM/TraSMAPI>
- <http://www.galirows.com.br/meublog/blog/tag/sumo/>
- [https://web.fe.up.pt/~niadr/PUBLICATIONS/2013/SUM02013\\_AnIntegratedFramework.pdf](https://web.fe.up.pt/~niadr/PUBLICATIONS/2013/SUM02013_AnIntegratedFramework.pdf)
- [http://agents.fel.cvut.cz/att2014/att2014\\_paper\\_16.pdf](http://agents.fel.cvut.cz/att2014/att2014_paper_16.pdf)
- [http://www.researchgate.net/publication/257807033\\_Using\\_TraSMAPI\\_for\\_the\\_assessment\\_of\\_multi-agent\\_traffic\\_management\\_solutions](http://www.researchgate.net/publication/257807033_Using_TraSMAPI_for_the_assessment_of_multi-agent_traffic_management_solutions)