

Autonomous Agents and Multiagent Systems

2nd Semester - 2015/2016

Agents In <Project Title>

Group 17 - <Taguspark>

*<André Lima>* - <70572>

*<Samuel Gomes>* - <76415>

*<André Fonseca>* - <84977>

# Abstract

The abstract of a document corresponds to a short and self-contained (maximum one page) description of the main aspects of the work presented. This summary plays a key role in motivating the reader to continue reading the entire document. The abstract should include the following aspects: (1) Motivation and description of the problem, *i.e.*, why is the work presented important and what is the problem you are trying to solve; (2) Approach, *i.e.*, what was the approach to solving the problem; (3) Results, *i.e.*, what is the proposed solution to the problem and the main results of the comparative study; (4) Conclusions, *i.e.*, what are the implications of the solution.

**Keywords:** a list separated by commas with no more than seven key words that characterize the work.

# Index

1. Introduction 4

2. The Scenario 4

3. Agent Architectures / Algorithms 4

3.1 Reactive Architecture 4

3.2 Deliberative Architecture 4

3.3 Hybrid Architecture 5

3.4 Coordination, Cooperation, Negotiation 5

3.5 Emotions Component 5

3.6 Learning Component 5

4. Comparative Study 5

5. Conclusions 6

6. References 6

# Introduction

In this section you should describe the purpose of the work, the problem that is being addressed, the approach and structure of the report.

**Notes:**

* The Introduction *is not* a copy of the abstract;
* Reports should not exceed 20 pages, including the cover, abstract, index and references, according to the format shown in this document.
* All references used in the development of the project must be clearly marked and placed in the References section.

# The Scenario

This section is especially important in projects where the realization of the environment is one of the objectives of the project, that is, for projects whose theme was proposed. The environment’s elements, dynamics, implementation details and possible existing parameterization for the same should be described and justified. The description of the agent’s perceptions, actions, possible limitations, etc. should also be provided.

For projects implementing one of the provided proposals, this description should only summarize the involved entities, developed sensors, considered perceptions and actions, and extra considerations.

Os agentes reativos têm duas perceções. Estas são os métodos “in-sight” e “in-corer”. Ambos os métodos são booleanos. O primeiro método devolve o valor lógico de verdade se a presa estiver dentro do campo de visão do agente que chamou esse método. O segundo método só é chamado se a presa estiver dentro do campo de visão do agente em questão e devolve o valor de verdade se a presa estiver num dos cantos do mundo.

Estes agentes têm também quatro atuadores, correspondentes aos quatro sentidos de movimento nos quais os agentes podem se mover (cima, baixo, esquerda e direita). Os agentes só se podem mover para zonas adjacentes a eles e que estejam desocupadas, isto é, sem outros agentes ou sem a presa.

# Agent Architectures / Algorithms

In this section you should describe the several architectures and algorithms according to what is required in the project’s description, *e.g.*, reactive, deliberative / BDI, hybrid, etc. The text should present a conceptual description mentioning aspects of implementation only if necessary for the understanding of the concepts. Some of the aspects to describe in each subsection are:

## Reactive Architecture

* The *<perception> \* → <actuator>* rules developed for each type of agent;
* The description of the arbitration mechanism of rules, *i.e.*, how they are ordered;
* The justification for any internal state variable that was used.

Para o problema proposto, os nossos agentes, os lobos, possuem um estado interno. Um campo de visão (field of view ou fov) que é, no máximo e em cada dimensão, metade do tamanho do mundo nessa dimensão. Este campo de visão é feito como um quadrado de dimensão d x d à volta do agente. Usando este estado interno, os agentes conseguem detetar se a presa está dentro do seu campo de visão. Usámos um número que representa d como estado interno, esse número foi usado como estado interno porque é uma característica dos agentes.

Em relação às regras perceção -> ação, estas foram desenvolvidas igualmente para todos os lobos.

Neste caso cada lobo verifica se a presa está no campo de visão, e tenta ir na sua direção.

Em último caso, quando algum lobo não consegue ver a presa, apenas escolhe aleatoriamente uma de cinco ações: andar em frente, para trás, para os lados ou ficar parado.

## Deliberative Architecture

* The beliefs of each type of agent;
* The desires of each type of agent;
* The intentions of each type of agent;
* How do beliefs originate desires and, lastly, the intentions;
* The kind of plans the agents create and follow in order to implement the intentions;
* Description of possible changes to the original BDI algorithm, justifying them.

No que diz respeito a esta parte do projeto, chegámos à conclusão que uma arquitetura BDI seria demasiada complexa para o tipo de agentes que considerámos.

Assim não seguimos a estrutura BDI dado que a implementação desta técnica se revelou opcional.

A nossa implementação resume-se a três módulos:

- A comunicação entre os agentes de modo a obterem um consenso sobre o seu sub-objetivo

- Procura de caminhos: recorremos ao algoritmo A Star, tal como na aula de laboratório 3, para procurar caminhos mais curtos até cada uma das posições adjacentes ao lobo. Assim, usámos como heurística para atualizar os custos, a distância euclidiana entre pontos dado que mapeamos a solução ótima ao caminho mais curto. Os nós da procura são implementados como uma lista com 3 campos que representam o f, o g e uma posição na grelha ( representada por uma lista com coordenadas )

- Estado Idle: quando um lobo não tem informação necessária para poder cumprir o seu objetivo, este entra num estado que consiste em vaguear pelo mundo em zigzag de modo a voltar a obter informação o mais rapidamente possível. Esse movimento é gerado a partir de operações locais que usam os limites do mapa para poderem mudar de direção quando conveniente. Esta operação têm em conta o field of view do agente, dado que quanto maior o fov do agente menor a frequência com que precisa de passar pelo centro do mapa (dado que adquire mais dados sobre o mundo a cada momento).

## Hybrid Architecture

* Description of the hybrid architecture, if implemented;
* How the mediation between the deliberative layer and the reaction was made.

## Coordination, Cooperation, Negotiation

In this section it should be described all coordination and/or cooperation algorithms and communication protocols used to create cooperative agents, if implemented in the project.

**Note:** This section is especially important for the project “*Warehouse Delivery”*. In this project, it should also be explained the algorithms for coalition formation (one subsection each) that were implemented.

## Learning Component

If learning techniques were used in the project, detail the aspects on which learning was incorporated, what learning algorithms were used and how they were implemented in the specific case of the project.

**Note:** This section is especially important for the project “*Wolf Pack”*.

# Comparative Study

In this section you should present the comparative study made between the several approaches explored in the creation of agents. In particular, the study should be able to provide conclusions about which of the approaches explored contributed more to the "success" of the agents.

Some of the aspects to describe this section are:

* The approaches that will be compared in the tests;
* The metrics used to determine the "success" of each agent / team;
* The tests that were designed;
* The results obtained in each studied condition (charts, tables, etc.);
* A statistical analysis of the results;
* Conclusions (limitations / advantages of each approach, etc.).

**Notes:**

* Any conclusions must be substantiated by the results of the experiments;
* This section applies to all projects, *i.e.*, even if the requested comparative study does not refer to the several architectures used, the general indications described above remain valid, only the object of the comparative study changes.

# Conclusions

This section should summarize the proposed solution to the problem presented in the Introduction. In particular, you should present an interpretation of the results described in the Comparative Study section.

**Note:** It is also expected in this section to describe how the work presented could be extended in order to motivate future work.

# References

This section should list all the references considered relevant to the understanding of the work presented. Here are some examples of reference formatting for several types of publication:

**For a book:**

1. D. Patterson and J. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, San Francisco, CA, USA: Morgan Kaufmann Publishers, 1997.

**For a journal article:**

1. B. Kernighan and S. Lin., An Efficient Heuristic Procedure for Partitioning Graphs. *The Bell System Technical Journal*, Vol. 49, No. 2, pp. 291-307, February 1970.

**For a paper included in the proceedings of a conference:**

1. F. Fallah, S. Devadas, and K. Keutzer. OCCOM: Efficient Computation of Observability-Based Code Coverage Metrics for Functional Simulation. In *Proceedings of the Design Automation Conference*, pp. 152–157, June 1998.

**For a dissertation:**

1. D. Cheng. *Power Estimation of Digital CMOS Circuits and the Application to Logic Synthesis for Low Power*. PhD thesis, University of California at Santa Barbara, December 1995.

**For a technical report:**

1. E. Sentovich. *SIS: A System for Sequential Circuit Synthesis.* University of California, Berkeley, April 1992.

**For a webpage:**

1. *Instituto Superior Técnico, Official webpage*. Retrieved from: <http://tecnico.ulisboa.pt/>. Last accessed April 2016.