

# Agentes Inteligentes aplicados a Jogos de Computador

Artur Ribeiro<sup>1[82516]</sup>  
a82516@alunos.uminho.pt

Universidade do Minho

**Resumo** No âmbito da disciplina de Agentes Inteligentes, no perfil de Sistemas Inteligentes, foi proposta a realização de um estado de arte sobre "Agentes e Multi-agentes".

O objetivo deste trabalho é perceber em que estado está atualmente esta área, abordando as suas propriedades e vertentes aplicando este paradigma a domínios concretos. Também será discutida as vantagens e desvantagens do uso de agentes e multi-agentes, fazendo uma análise de aplicações e projetos já existentes neste ramo e analisar a aplicabilidade e viabilidade destes conceitos aos jogos de computador como forma de os explorar mais aprofundadamente, melhorar as AI's dos jogos e a sua jogabilidade.

**Keywords:** Agentes Inteligentes · Ambiente · Gaming · AI · e-Sports.

## 1 Introdução

No âmbito da disciplina de Agentes Inteligentes, no perfil de Sistemas Inteligentes, foi proposta a realização de um estado de arte sobre "Agentes e Multi-agentes".

Com este trabalho, pretende-se fazer uma investigação sobre agentes e multi-agentes e a sua aplicação a domínios concretos, neste caso jogos de computador. Irão ser abordados os principais requisitos e funcionalidades que os agentes e os sistemas multi-agente (SMA) devem ter. Também serão descritas e avaliadas as principais vantagens e desvantagens dos diferentes tipos de agentes e SMA's. Por fim, terá uma análise crítica de algumas aplicações e projetos de investigação nesta área e analisará a aplicabilidade e viabilidade deste paradigma aos jogos de computador.

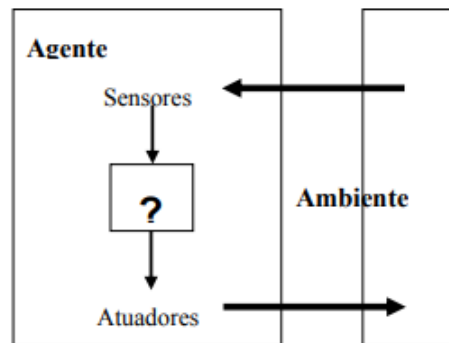
O tema dos jogos de computador surge devido ao interesse próprio que tenho nesta área, devido a ter jogado profissionalmente *League of Legends* durante 3 anos, tendo sido consultor de e-Sports para uma empresa e realizado alguns projetos nesta área a nível nacional.

Assim, estará contido neste relatório uma investigação feita em MOBA's, *Multiplayer online battle arena* e outro tipo de jogos, com uma opinião pessoal de como SMA's podem melhorar e ajudar no desenvolvimento de jogos em diferentes aspetos.

## 2 Agentes inteligentes e Sistema Multi-Agente

### 2.1 Descrição geral

O conceito de agente é algo genérico e bastante abrangente. Podemos ver um agente como algo que obtém informação e conhecimento do ambiente em que está inserido através de sensores, atuando sobre este através de atuadores. São entidades capazes de comunicar, se necessário, ativos num ambiente complexo e persistentes para alcançar um ou vários objetivos que tenham. São dotados de uma certa autonomia, sendo reativos e pro-ativos dependendo do estado atual do ambiente, reagindo com o que acha que é a melhor ação a realizar perante a informação que dispõe.



**Figura 1.** Relação entre o agente e o ambiente, através de sensores e atuadores.

De facto, sendo o agente descrito como algo complexo, propõe-se que este tópico seja acompanhado por um exemplo que ajuda na compreensão do tema de agentes inteligentes, mas também preparará o leitor para a entrada no mundo dos e-Sports.

Sendo este mundo algo mais recente que a noção de agentes, veremos o exemplo do xadrez, um jogo de tabuleiro que, apesar de ter regras simples, denota de um factor de inteligência bastante alto. Um bom jogador de xadrez (agente), terá de avaliar o risco da próxima decisão que tomará com o estado atual do tabuleiro para conseguir eliminar o rei do adversário, prevendo uma ou várias jogadas que o seu adversário poderá tomar para surpreender e tentar alcançar a vitória. É este mesmo sentimento de competição, estratégia e de risco, que obtemos também a jogar xadrez, que tornam os e-Sports algo fascinante e emergente, sendo que com a noção de agentes e multi-agentes inteligentes, atingiu-se um novo patamar na qualidade e jogabilidade dos jogos de computador, sendo este o tema principal deste relatório.



**Figura 2.** Exemplo de um tabuleiro de xadrez, semelhante a uma matriz.

Peça	Símbolo	Valor
Peão		1
Cavalo		3
Bispo		3
Torre		5
Rainha		9

**Figura 3.** Um tipo de estratégia comum que o agente pode adotar é quantificar as peças pelo seu valor. Note-se que o rei não aparece pois sem ele, perde-se o jogo.

Um sistema multi-agente não é mais que os agentes presentes nele. Estes, coordenam-se entre si para resolver um dado problema, ou vários, que sozinhos não conseguiriam resolver. A importância que os SMA's têm advem das propriedades e características dos seus agentes.

Deste modo, existem duas formas de coordenação entre agentes, competitiva e cooperativa, sendo que na competitiva, os dois agentes competem entre si para alcançar um interesse próprio, tendo informação a partir da negociação com outros agentes. Por outro lado, na cooperação, os agentes partilham o seu conhecimento e outras características de outros agentes para, em conjunto, resolver um dado problema ao qual não conseguiriam sozinhos.

De acordo com o que foi dito acima, os agentes e SMA's trazem uma maneira de perceber problemas complexos e aprender de que forma podemos agir sobre estes para os resolver. Em jogos de computador, um dos maiores problemas que surgiu com a evolução do ser humano e da experiência de muitos anos dos video jogos, foi conseguir desenvolver bots em jogos que se comportassem similarmente a um humano a jogar, ou pelo menos dessem "luta" contra um ser humano.

Em suma, um bom SMA (quantificado como forte), é aquele que apresenta características similares à do ser humano e capaz de aprender como resolver o problema para o qual foi destacado. No caso contrário, este é denominado de fraco, se não apresentar uma quantidade de características fundamentais para os SMA's. Algumas destas características irão ser apresentadas no tópico seguinte para perceber melhor o conceito de agentes e o quão abrangente este pode ser.

## 2.2 Características de agentes e SMA

Tendo então falado do conceito de SMA fortes e fracos, enumera-se algumas das mais importantes características que os agentes e SMA podem ter:

- **Autonomia:** O agentes agem sem ação de outros agentes, tendo controlo sobre as suas ações e sobre o seu conhecimento;
- **Reatividade:** São reativos, respondendo rapidamente aos dados sensoriais que recebem do ambiente;
- **Pro-atividade:** São pro-ativos, procurando novas formas de resolver o seu problema com iniciativa própria;
- **Sociabilidade:** Através da cooperação ou da competição;
- **Mobilidade:** Deslocam-se pelo ambiente tentando cumprir os objetivos que tem;
- **Intencionalidade:** Capacidade de definir objetivos ou estratégias;
- **Aprendizagem:** Capacidade de aprender, assimilando padrões ou ações que obtiveram melhores resultados;
- **Competência:** Conseguir resolver com sucesso os seus objetivos;
- **Veracidade:** O agente não fornece informação falsa, intencionalmente;
- **Racionalidade:** Racionam sobre os problemas que lhe são atribuídos, aceitando aqueles que forem possíveis realizar, que não tenham um quociente risco/reward muito elevado e que não vão contra os seus princípios.

- **Benevolência:** Ajuda outros agentes incorporando como seus, objetivos de outros, se forem em contra aos seus princípios;
- **Emotividade:** Arquiteturas como a BDI mostram uma similaridade entre o comportamento de um agente e um ser humano.

Estas características são apenas algumas das mais importantes características de agentes inteligentes. Algumas delas são contraditórias e não coexistem no mesmo agente, mas podem existir num SMA, sendo este uma mais valia para ambientes complexos em que a ação de mais que um agente seja necessária para alcançar os objetivos.

### 2.3 Tipos de Agente e Arquiteturas de Software

De acordo com as características que o agente possui e das ações realizadas por ele sobre o ambiente, podemos distinguir vários tipos de agentes, em relação à/ao:



- Mobilidade: Se o agente é móvel ou estacionário no ambiente;
- Realização de tarefas: agentes de serviço que realizam tarefas repetitivas, ou agentes pessoais que adquirem, organizam e disponibilizam informação para resolver tarefas;
- Modo de seleção de agentes:
  - Reativos: Baseiam-se em regras de condição/ação;
  - Memória: Baseiam-se em resultados obtidos no passado para tomar novas decisões, p.e Case based reasoning;
  - Orientado por objetos: Invocando procedimentos para obter informação pertinente e realizar ações;
  - Adaptativos: Alteram o seu comportamento de acordo com o estado do ambiente ao longo do tempo;
  - Úteis: Usados para tomadas de decisão entre objetivos ou numa situação de conflito;

Em suma, um agente deve ser capaz de aprender, agir sobre o ambiente e, de certa forma, evoluir ao longo do tempo, ficando mais inteligente. Assim, surgiram arquiteturas que descrevem a forma como o agente interage, se organiza e pelo seu modo de funcionamento, podendo ser deliberativas, reativas ou híbridas. A seguir, será feita uma descrição breve das arquiteturas, avaliando as suas principais vantagens e desvantagens.

**Arquitetura deliberativa** Normalmente, através de uma representação simbólica ou predicativa do ambiente, o agente deliberativo é usado para resolver problemas com uma certa ordem de prioridade, alterando o seu comportamento e conhecimento de acordo com a forma como raciocina sobre a informação retida do ambiente.

Apesar de apelativo, advêm da natureza desta arquitetura algumas desvantagens, das quais, ser bastante difícil transpor para regras simples de lógica ou matemática os problemas do ambiente do agente. Imaginemos um jogo atual em que há mudanças a cada segundo no ambiente e o agente teria de conter toda essa informação traduzida de forma simples, o que pode ser bastante difícil num ambiente tão complexo como este. Também existe a dificuldade de conduzir o método de raciocínio do agente, aliado à complexidade que alguns problemas podem ter e que nem mesmo nós sabemos qual a melhor maneira de raciocinar sobre eles. Em casos em que o problema possa ser totalmente transcrito de forma simbólica, esta arquitetura é bastante vantajosa devido à sua rapidez, funcionando pela lógica ou por cálculos matemáticos, tendo uma simplicidade enorme em relação a outras.



**Figura 4.** Exemplo simples de um agente deliberativo, retirado do powerpoint disponibilizado pela equipa docente.

**Arquitetura Reativa** Ao contrário da anterior, esta arquitetura não tem qualquer representação simbólica do ambiente, é apenas uma entidade capaz de agir sobre este. Tem objetivos a cumprir, percecionando o ambiente à sua volta através de sensores e reagindo sobre regras de condição/ação que lhe são impostas. Contudo, o seu comportamento é autónomo, sendo o agente a tomar decisões, cocorrentemente, de acordo com a perceção do ambiente que o rodeia.

Esta prespetiva pode ser vantajosa quando é possível descrever, na totalidade, todas as condições que acontecem num ambiente, pois o agente será rápido a reagir a esses eventos. Por outro lado, esta visão restringe o funcionamento do agente e limita a sua capacidade de aprendizagem em ambientes mais complexos, tendo pontos positivos e negativos.



4:

**Figura 5.** Exemplo simples de um agente reativo, retirado do powerpoint disponibilizado pela equipa docente.

**Arquitetura Híbrida** Devido às vantagens e desvantagens que as arquiteturas de SMA acima dispõe, surgiu a necessidade de criar as arquiteturas híbridas.

Estas arquiteturas são bastante mais adptáveis, robustas e desempenham melhores ações que as anteriores, sendo por exemplo usadas na robótica. Apresentam as características vantajosas de ambas as anteriores, tendo como principal ideia uma disposição hierarquica das funcionalidades do agente. Desde modo, dividem o agente com uma camada para a parte deliberativa, para a parte reativa e para de execução/planeamento de tarefas. É atribuída uma maior prioridade à camada reativa sendo que esta é importante para a resolução rápida de problemas, mas também contém a parte deliberativa que percebe o ambiente ao seu redor, planeando ações futuras.



**Figura 6.** Exemplo simples de um agente híbrido, retirado do powerpoint disponibilizado pela equipa docente.

**Arquitetura BDI** Esta é uma arquitetura deliberativa, contudo assenta sobre outros princípios. A arquitetura *Beliefs, Desires and Intentions* tem como principal objetivo uma descrição do conhecimento que o agente possui, através do estado mental do mesmo. Será sobre estes que ele se vai reger para agir no ambiente.

Podemos ver as crenças como algo que seja tomado como verdadeiro para o agente, os desejos como o estado de inteligência que o agente anseia ter e as intenções como a planificação e estratégia para conseguir realizar os seus desejos.

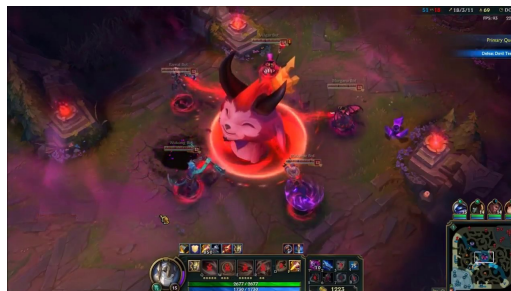




**Figura 7.** Exemplo simples de um agente BDI, retirado do powerpoint disponibilizado pela equipa docente.

No estado inicial dos jogos de computador, os bots não eram mais que uma sequência de condições "if", dando como exemplo o início do jogo League of Legends, que constará mais à frente no relatório como domínio a aplicar o conceito de agentes inteligentes. Ao longo dos anos, foi necessário evoluir o comportamento dos bots nos jogos de computador (tipicamente falada como a inteligência artificial) para acompanhar a evolução do ser humano neste ramo.

Foi assim que surgiu a ideia de investigar sobre os jogos de computador, avaliando a aplicabilidade e viabilidade dos SMA para melhorar os video jogos.



**Figura 8.** Doom bots, modo de jogo no League of Legends onde os bots são mais inteligentes e melhores em termos de tomadas de decisão, jogando 5 bots contra 5 jogadores humanos.

## 2.4 Áreas de aplicação, plataformas disponíveis e investigações na área

Tal como falado anteriormente, as áreas de aplicação de agentes e SMA é enorme e serão apenas anunciadas algumas áreas de interesse. Serão também analisadas algumas plataformas fundamentais para criar agentes e usar este tipo de paradigma, acompanhado por algumas investigações, já no domínio de interesse deste relatório, os video jogos.

Algumas áreas de aplicação de agentes que podemos ver perto de casa, destacam-se: área industrial, com o controlo de processos, produção automática e sistemas de controlo de tráfego aéreo. No setor comercial, o marketing, gestão de empresas e comércio tiveram algumas melhoras com o conceito de agentes. É de notar também o uso de agentes para a medicina, com a melhoria em análises de doentes, monitorização de pacientes, e no domínio do entretenimento, os video jogos.

Algumas ferramentas que permitem a construção de agentes e SMA, destacam-se duas, JADE e JESS. Ambas são ou irão ser utilizadas durante as aulas práticas e foi por isso mesmo que foram destacadas das restantes, não por serem melhores. JADE, ou *Java Agent Development Framework*, permite a criação de aplicações baseadas em agentes inteligentes. Esta ferramenta escrita em Java permite ao programador centrar-se apenas nas funcionalidades dos agentes e nos mecanismos inteligentes, não tendo que tratar da parte de envio de mensagens, codificação/descodificação de estruturas de dados, entre outras coisas. Já a ferramenta JESS, *Java Expert System Shell*, escrita também em Java, usa o algoritmo de Rete permitindo usar a memória para reter informação sobre o sucesso/insucesso no reconhecimento de padrões feita em tentativas anteriores. A sua maior vantagem é a portabilidade mas vem com um custo na eficiência do agente.

Já nas investigações, destacam-se algumas nas áreas dos video jogos, como o Projeto Soar/Games, em que o principal aspeto foi de construir um engine IA para jogos que podesse ser usado em vários jogos, especificando apenas particularidades de cada jogo. Os agentes no jogo seriam reativos, flexíveis, realísticos e adaptáveis. Esta investigação divide o engine em 3 processos: interface com o jogo, máquina de inferência e base de conhecimento, suportando toda a inteligência dos agentes. Este projeto foi aplicado aos video jogos devido à fácil percepção que podem ter, ao carácter visual apelativo e à forma como os jogos são baseados em ações para os completar.

Outra investigação nesta área foi em jogos MOBA, mais propriamente no League of Legends, testando a teoria de agentes inteligentes para construir um robo inimigo que jogasse tão bem como um jogador humano com alguma experiência. Foi utilizada a técnica de mapas de influência, que basicamente transforma áreas do mapa em valores positivos e negativos que influenciam a navegação e as ações do agente. Mais a frente irá ser abordada esta investigação pois tem o mesmo domínio de aplicação escolhido e explica bastante bem como agentes podem melhorar o *pathing* e as micro decisões que bots têm de tomar durante o jogo.

### 3 Jogos de Computador

Os video jogos, do tipo MOBA e RTS, *Multiplayer online battle arena* e *Real Time Strategy*, respetivamente, são jogos entre 1 ou vários jogadores, que têm algumas características interessantes que devemos ter em conta antes de abordar o domínio em questão.

Ambos os tipos de jogos acima referidos têm características em comum e bastantes diferentes de outros, tais como:

- Um ambiente determinístico, em que uma ação normalmente conduz ao mesmo *output*;
- Existe informação que não está disponível para o jogador durante algum tempo ou até ter visão sobre o inimigo;
- Os mapas do jogo podem ter a mesma configuração, mas não são sempre iguais. Não é guardado um estado do jogo para sempre no final de cada partida;
- São jogos com decisões em tempo real e com imensas características que podem ser tidas em conta ao longo do tempo, podendo também mudar.

A partir destas características, podemos ver que aplicar agentes inteligentes neste contexto pode trazer melhorias no pathing de bots, escolhendo os melhores locais que anteriormente trouxeram mais vantagens no jogo e que puseram o seu adversário numa posição pior que a que estava anteriormente. Como estes ambientes são determinísticos, podemos também utilizar o conceito de memória num agente para decidir as melhores ações a tomar em relação à área do jogo, estatísticas dos inimigos e da sua equipa, com base nas ações tomadas anteriormente (case based reasoning).

Por outro lado, também podemos perceber que, como estes jogos têm ambientes bastante complexos e dinâmicos ao longo do tempo, a escolha de um agente reativo também é importante. Este deve saber fazer escolhas, como coleccionar dinheiro, escolher atacar um inimigo ou fugir medindo a força, entre muitas mais decisões que têm de ser tomadas em tempo real. Isto torna difícil a tarefa de decisão entre diferentes *features* que o agente tem de tomar para ganhar o jogo e é o que torna interessante este conceito. Além disso, como o mapa, apesar de ter a mesma configuração (ver abaixo exemplo do mapa do *League of Legends*), os jogos não são sempre iguais. Os jogadores podem escolher diferentes personagens que fazem coisas diferentes, cada *game environment* também pode variar de jogo para jogo, mas contém sempre estruturas permanentes, que ajudam na implementação do nosso agente.



**Figura 9.** Exemplo do mapa do league of legends, *Summoner's Rift*, que sendo sempre igual em formato, difere em alguns monstros(bots) e em personagens escolhidas pelos jogadores, de jogo para jogo.

## 4 Conclusão

A grande dificuldade na aplicação a este tipo de domínios é a de conseguir estratificar que ações são mais prioritárias que outras, num ambiente em constante evolução. Para mim, a maior vantagem da aplicação de agentes é poder melhorar a experiência do jogador quando encontra inimigos bots no seu jogo, tendo pathings mais inteligentes e atacando de forma mais eficiente de acordo com as decisões que o jogador vai tomando contra eles.

Tal e qual como eu a jogar LOL, existe vários fatores que só no decorrer do jogo é possível saber se são vantajosos ou não e o agente terá o mesmo comportamento. Como a perspectiva de um agente se assemelha à perspectiva que um novo jogador tem ao começar a jogar este tipo de jogos complexos, os agentes inteligentes podem ser uma mais valia no desenvolvimento do jogo. Imaginemos que no decorrer da realização de um jogo são usados agentes para o testar, como foi feito em algumas das grandes empresas de jogos eletrónicos, podendo ser detetados vários bugs ou usar agentes para escolher a melhor maneira de um bot se comportar durante o jogo, para no lançamento o jogo estar perfeito.

Em suma, já é possível encontrar comportamentos inteligentes e adaptáveis em jogos de computador da atualidade, havendo sempre margem para melhorias. Como trabalho futuro, espero poder aplicar os conhecimentos de agentes e SMA, no âmbito da teoria de jogos.

## Referências

1. [Wooldrige, 2002] Wooldrige M., *An Introduction to Multiagent Systems*, John Wiley & Sons, ISBN 0 47149691X, 2002.
2. d’Inverno M., Luck M., *Understanding Agent Systems*, Springer, ISBN: 978-3540407003, 2003.
3. Wooldridge M., Jennings N., “Intelligent Agents: Theory and Practice”, *Knowledge Engineering Review*, 10 (2), pp. 115 152, 1995.
4. Kravari K, Bassiliades N., A Survey of Agent Platforms, *Journal of Artificial Societies and Social Simulation* 18 (1) 11, 2015.
5. [McKinsey, 1952] McKinsey J., *Introduction to the Theory of Games*, McGraw-Hill Book Co, 1952.
6. *Sebenta de Agentes Inteligentes*, Paulo Novais & Cesar Analide, Setembro de 2016.
7. YANG, P., HARRISON, B., & ROBERTS, D. L. 2014. Identifying patterns in combat that are predictive of success in MOBA games. *In Proceedings of Foundations of Digital Games*.
8. <https://euw.leagueoflegends.com/en-gb/>
9. *On the Development of Intelligent Agents for MOBA Games*, 2017.