

# Agentes Inteligentes

## Agentes e Sistemas Multi-Agente

Paulo Jorge Moreira Lima  
a89983@alunos.uminho.pt

University of Minho, Department of Informatics, 4710-057 Braga, Portugal  
16 de Novembro de 2020

**Resumo** Os Sistemas Multi-agente tal como a inteligência artificial têm vindo a transformar-se numa parte essencial do mundo em que vivemos, desde de pequenas aplicações mobile até projectos de larga escala como automação de veículos para uso humano. Neste artigo vão ser introduzidos alguns conceitos, exploradas as vantagens de sistemas multi-agente e referidas algumas aplicações dos mesmos em contexto real.

**Keywords:** Sistemas Inteligentes· Sistemas Multi-Agente· Industria· Aprendizagem

## 1 Agentes Inteligentes

A inteligência artificial tem vindo a aparecer cada vez mais na nossa vida, o desenvolvimento dos sistemas computacionais permitiu que esta área pudesse ser mais explorada e utilizada por toda a população.

Os **agentes**, no contexto da inteligência artificial, são entidades autónomo que tentam alcançar um objectivo através de um conjunto de dados que este recolhe. Um **agente** distingue-se pela sua capacidade de adaptação a situações imprevistas, sendo que um **agente** mais completo(inteligente) tem uma maior capacidade de adaptação o que permite alcançar os seus objectivos de um forma mais célere.

Um **agente** deve exibir um conjunto de características que o façam distinguir de outro sistema inteligente, esse conjunto de atributos é dado em:

- **Autonomia:** os agentes operam sem a intervenção de outros agentes, e têm controlo sobre as suas acções e o seu estado de conhecimento interno
- **Reactividade:** : os agentes têm percepção do que ocorre no seu universo de discurso e respondem adequada e atempadamente a mudanças ocorridas nesse ambiente
- **Pro-actividade:** os agentes são capazes de tomar a iniciativa, conduzindo as suas próprias acções segundo um comportamento que é dirigido por objectivos
- **Sociabilidade:** os agentes interactuam com outros agentes, comunicando com estes, competindo ou cooperando na resolução de problemas que entre-tanto lhes tenham sido colocados.

No entanto para que o mesmo seja considerado forte basta conter uma das seguintes características.

- **Mobilidade:** capacidade do agente se movimentar através da rede
- **Intencionalidade:** capacidade do agente para definir objectivos assim como de estratégias para os atingir
- **Aprendizagem:** capacidade do agente para adquirir conhecimento
- **Competência:** capacidade do agente conduzir com sucesso e eficiência as tarefas que é incumbido de realizar
- **Veracidade:** um agente não fornece, propositadamente, informação falsa.
- **Racionalidade:** um agente racional não aceita realizar tarefas que lhe pareçam impossíveis de executar, contraditórias com os seus princípios (e.g. ética), ou quando não são compensados em termos do risco, custo e esforço envolvido
- **Benevolência:** um agente benevolente adopta como seus, os objectivos de terceiros, desde que estes não entrem em conflito com os seus
- **Emotividade:** o agente apresenta certas características próprias do ser humano.

### 1.1 Arquitecturas para Agentes de Software

A arquitectura de um agente de software é definida pelas suas unidades funcionais e pela sua forma de organização e interacção. As arquitecturas que se seguem são algumas propostas para a criação de agentes com o objectivo de resolver os problemas encontrados na área da inteligência artificial.

- **Arquitectura Deliberativa:** Agentes deliberativos, possuem uma representação simbólica do mundo, sendo que as suas decisões são feitas por meio de um processo baseado em raciocínio lógico. Este raciocínio trabalha sobre um conjunto de símbolos que, sendo fisicamente concebíveis, podem ser combinadas formando-se estruturas sobre o qual se pode operar. Para decidir que acção tomar o agente precisa primeiro de conseguir representar simbolicamente o mundo envolvente, isto é, se um agente tem conhecimento de que uma de suas acções levarão a um de seus objectivos, então este vai seleccionar essa mesma acção.
- **Arquitectura Reactiva:** é um tipo de arquitectura mais simples que pretende responder a certas acções apenas com o formato (condição/acção). Numa arquitectura reactiva, o processo de tomada de decisão de um agente ocorre em tempo real, em resposta a estímulos do ambiente, captados pelos seus sensores, ou através de mensagens enviadas por outro agente. O mecanismo de controle pode então responder aos dados captados através do formato (condição/acção)
- **Arquitectura Híbrida** Arquitecturas híbridas surgem como tentativa de resolução das falhas encontradas nas arquitecturas deliberativas e reactivas, reunindo propriedades de ambas. As arquitecturas reactivas têm dificuldades em modificar os seus planos de acção a partir do momento em que a

situação passa a divergir de seus objectivos iniciais. No caso de arquitecturas deliberativas, situações imprevistas são a sua maior fraqueza. As arquitecturas híbridas devem definir agentes dotados de capacidades reactivas, de raciocínio e planeamento, resolvendo as limitações provenientes de outras abordagens.

- **Arquitectura BDI:** Este tipo de arquitectura considera três estados mentais: Crença, Desejo e Intenção. Estas arquitecturas surgem através do processo de decisão, momento a momento, tentando obter a acção a desempenhar com vista em alcançar os seus objectivos. Este processo envolve a decisão de objectivos a atingir e qual a melhor forma de atingir esses objectivos. A crença que um agente tem é a sua representação do mundo. As intenções são acções definidas com base nos desejos. A mudança da crença de um agente pode modificar todos os outros estados sendo que se baseiam uns nos outros. Este tipo de abordagem é ignora os factores emocionais do ser humano aplicados ao agente, pelo que pode ser considerado reducionista.

## 2 Sistemas Multi-Agente

Um Sistema Multi-agente é na sua essência um conjunto de entidades que cooperam de forma a solucionar um dado problema que normalmente exige uma cooperação e utilização de todas as suas capacidades individuais.

Em geral todos os sistemas multi-agente devem seguir alguns conceitos:

- **autonomia:** os agentes são activos e responsáveis pelas suas próprias actividades e as decisões a tomar
- **encapsulamento:** um agente não tem uma visão global do sistema, ou um sistema deve ser demasiado complexo para que o agente possa influenciar negativamente o mesmo
- **descentralização:** o agente não tem controlo total do sistema, isto é, existe uma separação do poder entre o mesmo. Caso este não seja o caso, então o sistema pode ser reduzido a apenas um agente.

Um sistema multi-agente faz uso das capacidades individuais de cada agente e tira proveito do relacionamento que cada um tem entre si, este relacionamento deve ser coordenado e se assim o for permite que o sistema consiga alcançar os seus objectivos.

As vantagens dos sistemas multi-agentes são principalmente a resolução de problemas de grande escala, que possibilitam através da autonomia dos agentes, uma maior estabilidade e confiança no sistema.

## 2.1 Linguagens de Comunicação entre Agentes de Software

A comunicação é um dos mais importantes factores dentro dos sistemas multi-agente, como tal, para facilitar o desenvolvimento dos mesmos, foram criadas plataformas que auxiliam na criação dos relacionamentos entre agentes:

- **DCOM/OLE:** A plataforma DCOM/OLE (Distributed Component Object Model/Object Linking Embedding) da Microsoft privilegia a interoperabilidade entre objectos desenvolvidos segundo diferentes linguagens de programação, tendo em vista os sistemas operativos Windows.
- **JATLITE:** A plataforma JATLite (Java Agent Template Lite) foi desenvolvida na Universidade de Stanford, com o objetivo de facilitar a comunicação entre agentes heterogéneos e distribuídos. A comunicação é realizada através da troca de mensagens. Qualquer agente do sistema deve registar-se no AMR, usando um nome e uma palavra-chave. Este AMR gere o processo de encaminhamento de mensagens do emissor e entrega-as ao receptor, mesmo quando existem falhas de comunicação. O AMR mantém, por outro lado, uma actualização dos endereços dos agentes do sistema.
- **CORBA:** A plataforma CORBA (Common Object Request Broker) tem objectivos similares as plataformas referidas anteriormente. Existe um alto nível de abstracção que permite manter um certo encapsulamento. A comunicação entre aplicações é realizada pelo envio, para a arquitectura CORBA, de uma descrição na linguagem IDL, do serviço pretendido. É da responsabilidade da arquitectura CORBA encontrar a aplicação que satisfaz os requisitos desse pedido, realizar o pedido e retornar o resultado à aplicação cliente.).

## 2.2 Ferramentas de Desenvolvimento de Sistemas Multi-agente

Para que seja possível fazer um desenvolvimento rápido e correto é necessário o auxilio de plataformas que permitam gerir os agentes que sejam criados, manter a comunicação entre estes.

Existem várias ferramentas para o desenvolvimento de sistemas multi-agente sendo que as seguintes JADE, Aglets, Cougaar, JACK.

O **JADE** (Java Agent Development framework) é uma free software framework utilizada para o desenvolvimento de aplicações de agentes em conformidade com as especificações FIPA para sistemas multi-agentes inteligentes interoperáveis. O JADE é escrito em linguagem Java e é feito por vários pacotes Java, fornecendo aos programadores peças prontas de funcionalidades e interfaces abstractas para tarefas personalizadas. As principais ferramentas fornecidas pelo JADE são o Remote Management Agent (RMA), o Dummy Agent, o Sniffer Agent, o Introspector Agent, o SocketProxyAgent, a GUI do DF (uma user interface gráfica completa usada pelo Diretory Facilitator).

**Aglets** é uma plataforma baseada em Java e biblioteca para construir aplicações baseados em agentes móveis. Um aglet é um agente Java que se pode mover de

forma autónoma e espontânea de um host para outro, carregando um pedaço de código com ele. Ele pode ser programado para executar em um host remoto e mostrar diferentes comportamentos em diferentes hosts. As implementações de segurança baseadas em Java cuidam do acesso autorizado a recursos locais nos hosts remotos.

O **Cougaar** (acrónimo inglês de Cognitive Agent Architecture) é uma plataforma de desenvolvimento de aplicações baseadas em agentes. Foi implementada em linguagem Java e resulta de um projecto de investigação DARPA destinado a disponibilizar um plataforma open source flexível para o desenvolvimento de aplicações baseadas em agentes de diferentes tipos.

A ferramentas **JACK** Agentes inteligentes é uma estrutura em Java para o desenvolvimento de sistemas multi-agentes. Os aplicativos JACK consistem numa coleção de agentes autónomos que obtêm informações do ambiente e comunicam com outros agentes. Isto faz com que existe um forte encapsulamento. Cada agente é definido em termos de seus objetivos, conhecimento e capacidade social, sendo então deixado para desempenhar a sua função de forma autónoma dentro do ambiente em que está inserido. JACK é um dos poucos sistemas multi-agentes que usa o modelo de software BDI e fornece sua própria linguagem de plano baseada em Java e ferramentas de planeamento gráfico.

### 3 Aplicações de Sistemas Multi-Agentes

Os sistemas multi-agentes são aplicados em várias áreas entre as quais, na indústria dos videojogos, optimização de logística e produção, reconhecimento de padrões com utilização na medicina ou produtos diversos, visão por computador, reconhecimento de voz, controlo adaptativo de processos e muitos mais. Na área da saúde existem vários exemplos de Sistemas Multi-Agentes como o TeleCARE e o MADIP.

#### **TeleCARE**

O projecto TeleCARE visa a concepção e desenvolvimento de um framework configurável focado em comunidades online para apoio a idosos. Essas comunidades virtuais permitirão que os idosos fiquem em casa e mantenham, até certo ponto, seu estilo de vida típico. Os autores propõem uma solução que deve ser vista como complementar a outras iniciativas de integração dos idosos na sociedade. O sistema TeleCARE propõe o paradigma multi-agente como base de infraestrutura em vez de TCP/IP sobre a Internet por duas razões principais:

- Mover o código para o local onde as acções são necessárias permite resposta em tempo real, autonomia e continuidade da prestação do serviço com dependência reduzida da disponibilidade da rede e atrasos;
- Uma vez que novos agentes móveis podem ser construídos e enviados para execução remota sempre que necessário, níveis mais elevados de flexibilidade e escalabilidade são alcançados. Com diferentes organizações a desenvolver diferentes produtos e serviços, numa grande variedade de áreas, há uma necessidade de uma plataforma comum na qual todos esses desenvolvimentos possam ser conectados para que a interoperabilidade seja possível. Como tal, o TeleCARE tenta ser o mais aberta possível para que novos dispositivos se possam integrar com a plataforma à medida que são desenvolvidos.

#### **MADIP**

A combinação de tecnologias existentes, como Redes de Sensores Sem Fio (RSSF) e Redes de Sensores de Área Corporal (BASN) num sistema coerente, é chamada de rede de sensores corporais unificada. Um exemplo disso mesmo é a Multi-Agency Data Integration Project (MADIP). O sistema resultante é capaz de notificar o prestador de cuidados responsável sobre a anormalidade automaticamente, oferecer aconselhamento médico à distância e monitorizar de forma continua a saúde daqueles que precisam dele. Como tal, esta plataforma é adequada para a vida diária, bem como aplicativos de detecção e assistência para idosos.

## 4 Conclusão

Ao longo deste artigo foi feita uma introdução e estudo sobre a definição de agente e multi-agente como o objectivo de resolver problemas. Cada agente é uma entidade isolada, mas que consegue comunicar e utilizar essa mesma comunicação de forma a contribuir para um objectivo mais global do sistema em que se encontra, este beneficia do uso de vários agentes de forma a utilizar a sua escalabilidade para alcançar todos os seus objectivos

O conceito de agente não é consensual, no entanto assume-se que um agente deve ter um comportamento que permita reagir a certos eventos de uma forma autónoma e comunicativa, sendo que a capacidade de adaptar e diferentes situações define o seu nível de complexidade e inteligência. A forma como cada agente manipula todos estes factores determina como a arquitectura deste vai estar definida.

Foi também possível perceber em que áreas é que este tipo de sistemas é utilizado e também quais os exemplos de casos particulares.

Com este estudo foi possível perceber a vantagem do uso de sistemas multi-agente em comparação com agentes isolados e também expandir o conhecimento em relação ao seu funcionamento. Sistemas descentralizados expandem por isso as potencialidades já introduzidas anteriormente sobre agentes o que permite uma evolução muito mais rápida da tecnologia.



## Referências

1. PAULO NOVAIS, CESAR ANALIDE, *Agentes Inteligentes*, Universidade do Minho (Setembro de 2006)
2. PAULO LEITAO, Artificial Intelligence and Computer Science Laboratory  
<https://bibliotecadigital.ipb.pt/bitstream/10198/9677/1/BeyondAI-v1.pdf>
3. WIKIPEDIA,  
<https://en.wikipedia.org/wiki/Aglets>  
Last accessed 14 Nov 2020
4. WIKIPEDIA,  
[https://en.wikipedia.org/wiki/JACK\\_Intelligent\\_Agents](https://en.wikipedia.org/wiki/JACK_Intelligent_Agents)  
Last accessed 14 Nov 2020
5. MURILO JUCHEM E RICARDO MELO BASTOS, FACULDADE DE INFORMÁTICA  
<https://www.pucrs.br/facin-prov/wp-content/uploads/sites/19/2016/03/tr013.pdf>
6. WIKIPEDIA,  
[https://en.wikipedia.org/wiki/Intelligent\\_agent](https://en.wikipedia.org/wiki/Intelligent_agent)  
Last accessed 14 Nov 2020
7. ELHADI SHAKSHUKI, MALCOLM REID,  
<https://www.sciencedirect.com/science/article/pii/S1877050915008716>
8. JADE, *JAVA Agent DEvelopment Framework*,  
<https://jade.tilab.com/>  
Last accessed 12 Nov 2020
9. AOS GROUP, *AOS Group*,  
<https://aosgrp.com/products/jack/>  
Last accessed 15 Nov 2020
10. WIKIPEDIA,  
[https://pt.wikipedia.org/wiki/Sistema\\_multiagente](https://pt.wikipedia.org/wiki/Sistema_multiagente)  
Last accessed 14 Nov 2020
11. WIKIPEDIA,  
[https://en.wikipedia.org/wiki/Multi-agent\\_system](https://en.wikipedia.org/wiki/Multi-agent_system)  
Last accessed 14 Nov 2020