



**Ciências
ULisboa**

**Departamento de Informática da Faculdade de
Ciências da Universidade de Lisboa**

Sistemas Multiagentes 2022/2023

Projeto Final Gestão de Elevadores

Grupo 1

Aluno	Número
José Siopa	fc60716
Martim Costa	fc60504
Tomás Ferreira	fc59449

30 de janeiro de 2023

ÍNDICE

1	INTRODUÇÃO	1
2	OBJETIVOS	2
3	AGENTES E SEUS COMPORTAMENTOS	3
3.1	Agente elevador simples	3
3.2	Agente elevador cooperativo.....	3
3.3	Agente elevador com planeamento	3
4	DEFINIÇÃO DAS VARIÁVEIS GLOBAIS.....	4
5	CRIAÇÃO DO SIMULADOR E IMPLEMENTAÇÃO DOS AGENTES.....	4
5.1	Interface Gráfica.....	4
5.2	Agente Simulador.....	5
5.3	Agente elevador cooperativo.....	5
5.4	Agente elevador com planeamento	5
6	COMPARAÇÃO DO DESEMPENHO DOS AGENTES	6
7	CONCLUSÕES	7

1 Introdução

Este relatório aborda a tecnologia de sistemas multiagentes e a sua aplicação na criação de uma simulação de elevadores inteligentes.

Os sistemas multiagentes são sistemas complexos compostos por múltiplos agentes autónomos que trabalham juntos para alcançar um objetivo comum. Os agentes podem ser softwares, robôs, dispositivos ou até mesmo pessoas. Estes sistemas são projetados para agir de forma autónoma e adaptativa, com capacidade de tomar decisões e aprender com o seu ambiente.

A utilização desta tecnologia numa simulação de elevadores inteligentes permite testar diferentes estratégias de alocação para minimizar o custo de operação dos elevadores, entregar as pessoas no mínimo tempo possível e avaliar a sua eficiência.

2 Objetivos

Neste projeto tem-se como objetivo realizar um sistema multiagente para a gestão eficiente de elevadores num edifício.

Para tal, tomar-se-ão os seguintes passos:

- Especificar os agentes a serem criados e seus comportamentos no sistema
- Definir variáveis globais ao sistema
- Criação do simulador, implementando os agentes definidos
- Comparar os agentes, relativamente ao seu desempenho no sistema e às suas estratégias de cooperação

3 Agentes e seus comportamentos

Nesta secção são apresentados os diversos agentes desenvolvidos bem como as suas estratégias de cooperação entre outros.

3.1 *Agente elevador simples*

Este tipo de agente não tem qualquer componente de cooperação e sempre que for feito um pedido para algum andar, vai ser ele que vai responder ao pedido, não sendo capaz de delegar o pedido a mais nenhum agente.

3.2 *Agente elevador cooperativo*

Este tipo de agente, semelhante ao elevador simples, responde sempre que receber um pedido, mas também é capaz de delegar tarefas para outros agentes, de forma a distribuir carga pelos outros agentes e agilizar o processo de resposta a pedidos.

Esta delegação é feita após uma comparação de forma a saber qual o agente que se encontra mais perto do pedido. Todos os elevadores conhecem as posições finais dos outros agente.

Os agentes enviam a sua posição para outros elevadores ao terminarem o seu pedido e com essa informação é possível detetar qual o agente que se encontra mais próximo do pedido, fazendo o módulo da distância do piso atual dos agentes com o piso inicial do pedido, o elevador com o menor valor será o que se encontra mais perto. Em caso de distâncias iguais o próprio agente responderá a este pedido.

Para além da adição da componente de cooperação este agente também permite a receção de novos pedidos durante a viagem, ou seja, se receber um pedido que deseja ir na mesma direção que o pedido a que está a responder e se encontra num piso inferior do piso final do primeiro pedido, este novo pedido será adicionado aos pedidos que se encontra a responder.

3.3 *Agente elevador com planeamento*

Este tipo de agente, semelhante ao elevador cooperativo, vai possuir todas as capacidades já mencionadas para o agente cooperativo, mas vai também possuir uma peculiaridade. Este agente vai ser capaz de produzir planos locais de forma a tentar otimizar o tempo de resposta a pedidos.

Para definir o plano nós colocamos os pisos a partir dos quais são feitos mais pedidos e colocamos esses pisos numa matriz, juntamente com as frequências com que esses pisos são chamados. Desta forma, o agente consegue perceber quais são os pisos a que este mais responde e, quando este fica um determinado intervalo de tempo sem receber pedidos, este vai-se deslocar para o piso que mais é chamado.

Cada agente irá fazer o seu próprio planeamento (não existindo um plano partilhado entre todos os agentes), e este planeamento é acionado assim que o elevador passa dos estados “Moving Up” ou “Moving Down” para o estado “standby”, terminando a ação de planeamento e começando a ação de responder a pedidos, assim não interrompendo o comportamento principal do agente.

É também de notar que, desta forma, num cenário em que o número de pisos corresponde ao número de elevadores, poderão ficar alguns pisos sem elevadores, uma vez que estes vão migrar para os pisos mais chamados.

4 Definição das variáveis globais

Para garantir uma maior fidelidade na posterior comparação do desempenho dos agentes, é necessário garantir que certas variáveis entre cada simulação permaneçam iguais. Definiu-se então que, para todos os agentes, os elevadores movem-se à mesma velocidade, a capacidade máxima de cada elevador e o estado do elevador.

5 Criação do simulador e implementação dos agentes

Estando definidos os tipos de agentes e as variáveis, prosseguiu-se à criação do simulador e todos os seus componentes.

5.1 Interface Gráfica

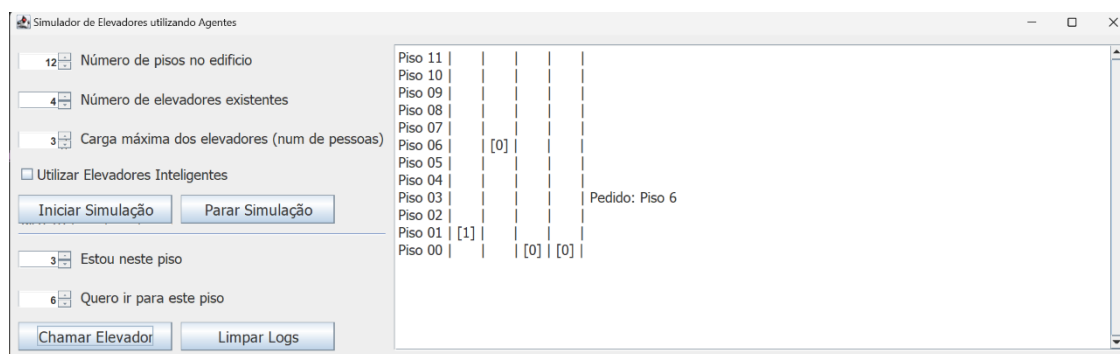


Figura 1 - Interface gráfica da aplicação

Para facilitar a criação de cenários de teste e visualizá-lo à medida que acontecem alterações, criou-se uma interface gráfica com recurso à ferramenta *Swing*. Esta interface apenas serve para a representação humana e não tem qualquer interferência no desempenho do sistema. Também permite definir alguns valores iniciais como o número de pisos, o número de elevadores e a lotação de cada um deles. É também possível enviar pedidos através da interface, em adição aos automáticos já existentes.

Como se pode ver pela figura 1, existem vários campos para introduzir valores. Acima da linha separadora, estão os campos que têm de ser introduzidos antes de iniciar a simulação, sendo estes o número de pisos, o número de elevadores, a carga máxima de cada um e que tipo de agente utilizar, com ou sem planeamento. Do lado direito, irá ser criado um edifício com estes valores. Caso algum valor não seja introduzido corretamente, a simulação não iniciará.

Na parte de baixo do separador, é possível o utilizador também enviar pedidos para os elevadores, onde estes vão ser representados no lado direito à frente do respetivo piso de seleção.

Os elevadores são representados por “[x]” e o valor “x” indica o número de pessoas dentro desse mesmo elevador.

No decorrer da simulação, ir-se-á observar pedidos novos a chegar e os elevadores a mexerem-se ao longo do prédio.

5.2 Agente Simulador

Este agente serve apenas como auxílio para permitir o funcionamento automático do sistema e a visualização do estado atual dos agentes.

O agente simulador possui um *TickerBehaviour* que envia pedidos aos agentes de x em x tempo, simulando pessoas em andares aleatórios a chamar o elevador para um andar aleatório.

Também possui um *CyclicBehaviour* que recebe qualquer mensagem enviada pelos agentes elevadores, para poder enviar os valores mais atualizados à interface gráfica e visualizar em tempo real as atualizações do sistema.

5.3 Agente elevador cooperativo

O agente cooperativo vai ter um *CyclicBehaviour* que está constantemente a receber *ACLmessages* e coloca-as numa *blocking queue*. Se receber uma mensagem, ela vai ser adicionada à *blocking queue* e vai iniciar um *OneShotBehaviour*. Este *Behaviour* vai ser responsável por retirar um elemento da *blocking queue* e em seguida vai fazer uma verificação.

Se o estado dele for *standby* e o pedido que ele recebeu for do tipo *ACL-Request*, o agente vai prontamente responder a esse pedido, fazendo também verificações se existe mais alguma pessoa que queira subir ou descer, dependendo do pedido.

Se o pedido que recebeu é do tipo *ACL-Inform*, então ele vai atualizar a sua *HashTable* com as posições dos demais elevadores.

Se o estado do agente não for *standby* e receber um pedido do tipo *ACL-Request*, então ele vai colocar esse pedido numa lista de pedidos auxiliar para ser atendido mais tarde.

À medida que o elevador vai subindo ou descendo andares, ele verifica se existe algum pedido na lista de pedidos auxiliar e, se houver, ele vai verificar se este pode ser atendido e, se poder ser atendido, vai colocar esse pedido na lista de pedidos que está a satisfazer atualmente.

5.4 Agente elevador com planeamento

O agente com planeamento vai ter os mesmos *Behaviours* que o Agente Elevador Cooperativo e mais um adicional.

Este agente vai possuir também um *OneShotBehaviour* que vai ser responsável por mover-se, passado um intervalo de tempo, para o andar de origem mais chamado.

Este *Behaviour* pode ser interrompido quando um novo pedido é recebido.

O *CyclicBehaviour* que está encarregue por receber os pedidos vai também colocar numa *HashTable* os pedidos de origem mais pedidos.

6 Comparação do desempenho dos agentes

Através do estudo realizado pelo teste de todos os diferentes tipos de agente abordados neste projeto, conseguimos perceber que, como esperado, o sistema de elevadores que possui planeamento teve um desempenho superior aos demais sistemas.

Devido ao facto de este possuir a capacidade de fazer planos e conseguir ter uma melhor percepção do meio ambiente, este foi capaz de se adaptar aos pedidos que lhe eram feitos.

Conseguimos também perceber que existe um *threshold* a partir do qual se percebe que os agentes que recorrem a planeamento se tornam mais eficientes.

Na tabela seguinte vamos apresentar os tipos de elevador e o cenário em que tornaria a sua utilização eficiente.

Tipo de agente	Dimensões do prédio e N de elevadores	Tempo de resposta
Elevador simples	<10 andares e <6 elevadores	O tempo de resposta vai ser o tempo que o elevador demorar a atingir o piso do pedido.
Elevador com cooperação	<10 andares e <6 elevadores, com um mínimo de 2	O tempo de resposta vai ser o tempo que o elevador mais próximo demorar a chegar ao piso pedido
Elevador com planeamento	Entre 10 e 20 andares e >6 elevadores	O tempo de resposta vai ser semelhante aos demais no início do uso dos elevadores, mas, à medida que o elevador vai aprendendo o padrão de uso, vai conseguir reduzir o tempo de resposta.

7 Conclusões

Com este projeto conseguimos compreender melhor como a cooperação entre agentes consegue melhorar o desempenho de um sistema multiagente.

Também conseguimos consolidar os nossos conhecimentos sobre a implementação de planos e como estes afetam os comportamentos dos agentes, bem como a sua performance.

Através da utilização de planos, vimos um aumento no tempo de resposta e desempenho do sistema, sendo que este iria reagir mais rápido aos pedidos do simulador.