



FEUP
Universidade do Porto
Faculdade de Engenharia

Agentes e Inteligência Artificial Distribuída

Relatório Intercalar

Escalonamento de Pacientes em Hospitais

4º ano

1º semestre

Mestrado Integrado em Engenharia Informática e Computação

Elementos do Grupo:

André Regado 201201757 - ei12182@fe.up.pt

António Presa 201204946 - ei12092@fe.up.pt

Francisco Couto 201206349 - ei12189@fe.up.pt

Novembro de 2015

Índice

Índice

Enunciado

Descrição do Trabalho

Objetivos do Trabalho

Resultados esperados e forma de avaliação

Plataforma/Ferramentas

Para que serve / Descrição das características principais

Realce das funcionalidades relevantes para o trabalho

Especificação

Identificação e caracterização dos agentes

Protocolos de interação

Faseamento do projecto

Recursos

Bibliografia

Software

Enunciado

- Descrição do Trabalho

O nosso trabalho consiste em construir um sistema multi-agente para o escalonamento de pacientes em hospitais. Um hospital é constituído por várias secções, distintamente geridas, que são visitadas por pacientes cujos tratamentos e exames dependem da sua doença.

Num sistema deste tipo, os agentes precisam de ser flexíveis à mudança, visto que, o tempo que os pacientes gastam nos hospitais é um problema que é confrontado, com uma grande percentagem de incertezas.

Os pacientes chegam constantemente ao hospital e a necessidade de tratamento médico nem sempre é determinado no início do processo de tratamento: novas descobertas durante os exames de diagnóstico podem mudar a prioridade dos pacientes, podem exigir tratamentos adicionais ou exames. Também o tempo de serviço para o tratamento e exame não são iguais em todas as pessoas. Complicações surgem e a chegada de pacientes com emergência resulta num distúrbio nos horários.

Para tentar resolver este problema, os hospitais começaram uma técnica semelhante à *first-come-first-served*. Desta forma, uma ala do hospital receitava os exames e tratamentos a cada paciente originando *requests* para as alas auxiliares que, responsáveis por administrar os tratamentos, chamariam os pacientes conforme a sua disponibilidade no momento.

Esta técnica apesar de ser útil para responder rapidamente a mudanças repentinas e a necessidades urgentes de atendimento, não permitia uma comunicação inter-unidade o que poderia levar a que duas alas requisitassem ao mesmo tempo o mesmo recurso/paciente.

Por existir esta restrição para cada ala poderiam ocorrer tempos de espera desnecessários e desagradáveis, bem como tempos excessivos de utilização dos recursos hospitalares.

- **Objetivos do Trabalho**

Tal como explicado anteriormente a tarefa que os hospitais têm de alocar recursos e pacientes é extremamente complexa, pelo que o objectivo seria arranjar uma solução otimizada que permitisse não só reduzir o tempo de espera de cada paciente, mas também permitir, por isso, uma alocação e utilização ótimas dos recursos disponibilizados para a aplicação de tratamentos e para a realização de exames necessários ao diagnóstico das mais variadas doenças. Para isso seria necessário implementar um sistema multi-agente e um mecanismo de coordenação que permitisse correlacionar eficientemente estes agentes.

- **Resultados esperados e forma de avaliação**

É esperado que o nosso sistema seja capaz de gerir um conjunto de recursos e pacientes num hospital alocando-os de uma forma otimizada e reduzindo tempos de espera (de ambos os recursos), bem como, caso seja necessário, lidar com situações esporádicas / urgência, originárias de novos tratamentos a pacientes já existentes, ou de pacientes que chegam gravemente doentes.

Plataforma/Ferramentas

- Para que serve / Descrição das características principais

O JADE é uma *framework* de software desenvolvida pela *Telecom Itália* em parceria com a Universidade de Parma, sendo atualmente um projeto *open source* com licença LGPL (*Lesser General Public Licence*).

Totalmente escrito em Java. É interoperável, uma vez que permite a comunicação entre agentes JADE com outros tipos de agentes fora da plataforma. Uniforme e portátil já que faz uso homogêneo de várias *APIs* independentes da versão Java.

É um sistema distribuído, onde os agentes habitam e possuem como forma básica de comunicação as mensagens assíncronas; baseado no paradigma *peer-to-peer*. A estrutura dessas mensagens é baseada na linguagem ACL (*Agent Communication Language*) do padrão *FIPA* que contém campos como contexto da mensagem e o tempo limite de espera pela resposta da mensagem. Os agentes são identificados por um nome global único.

O *Repast* (*Recursive Porous Agent Simulation Toolkit*) é um de muitos conjuntos de ferramentas de modelação de agentes que estão disponíveis. O seu grande objectivo passa por ir além da representação de agentes como entidades discretas e fechadas para uma visão de atores permeáveis e interligados.

- Realce das funcionalidades relevantes para o trabalho

O Jade será importante para a estruturação do sistema multi-agente e das várias características inerentes a cada um.

O *Repast* tratará de disponibilizar uma *GUI* de manipulação das várias características que compõem o sistema, bem como garantir uma interface de simulação *user-friendly* que demonstre a interação entre os vários agentes.

Especificação

- Identificação e caracterização dos agentes

- Paciente: Agentes que representam os pacientes do hospital e que necessitarão dos vários recursos de acordo com as suas necessidades. Poderão apenas ter acesso à sua própria agenda e serão caracterizados por uma **utility function** que permitirá atribuir uma cardinalidade ao seu estado de saúde. Essa *utility function* terá em consideração um custo de oportunidade de não curar de imediato a doença e, por conseguinte, o desenvolvimento do estado de saúde se não for administrado o tratamento.

Esta função será avaliada de 0 a 1 (0 - morte | 1 - saudável) e pode ser de uma forma comum denominada como "*years of well being*".

É caracterizado por uma prioridade, ou seja, o seu nível de emergência, por um nome, por um sintoma, por uma hora de chegada e pelo tipo de exames a efetuar.

-Recursos: Agente que representa os recursos hospitalares necessários para a realização dos exames do agente Paciente. É caracterizada pelo tipo de exame que realiza e possui enfermeiros associados, ou seja, técnicos responsáveis por realizar o exame. Cujo o objetivo é minimizar o seu tempo de inutilização.

-Hospital: Está encarregado de receber e processar a informação do agente do Paciente. É responsável por monitorizar o funcionamento do hospital, como atribuir um recurso a um paciente.

É caracterizado por um conjunto de pacientes, salas (que cada sala representa um recurso), médicos e enfermeiros.

Os nossos agentes dever-se-ão relacionar entre si da seguinte maneira:

1º- Introdução dos dados do paciente

2º- Introdução do estado do paciente

3º- Criar o agente paciente

4º- Adicionar uma nova entrada no agente hospital

5º- Agente hospital está constantemente a distribuir agentes pacientes pelos agentes recursos.

6º- Agentes recursos estão sempre a comunicar com os próprios para saber se há vagas noutros agentes, ou menor fila de espera. Se houver prioridade passar à frente

7º- Agentes recursos vão completando exames

8º- Agentes pacientes podem cancelar exame

- Protocolos de interação

Protocolos de Negociação- Agentes podem estabelecer acordos para cooperarem. Os protocolos consistem em procedimentos a seguir para conseguir um comportamento conjunto desejado levando ao encontro conjunto (aqui um contrato)

Os protocolos de negociação permitem entre outras coisas leilões.

No caso deste sistema multi-agente **competitivo** será extremamente importante este tipo de protocolo visto que para a atribuição de cada recurso do hospital a um paciente será realizado uma espécie de leilão inglês em que:

- compradores submetem propostas iniciais (valores desejados)
- leiloeiro dá a conhecer a melhor proposta
- compradores oferecem propostas crescentes
- (limites = valor da função de valor de cada agente)
- termina quando existe apenas uma proposta

- Faseamento do projecto

Primeiro, iremos começar por definir todas as classes necessários para o projeto e perceber como se irão ligar.

De seguida, iremos criar os agentes e fazer com que eles se relacionem corretamente.

Por fim, se tivermos ainda tempo e se for conveniente, iremos fazer a parte gráfica do problema.

Recursos

- Bibliografia

1. Paulussen, T., Zöller, A., Rothlauf, F., Heinzl, A., Braubach, L., Pokahr, A., Lamersdorf, W.: Agent-based patient scheduling in hospitals. In: P.L.O.S. S. Kirn O. Herzog (ed.) Multiagent Engineering - Theory and Applications in Enterprises, pp. 255-275. Springer (2006)
2. <https://www.activecomponents.org/bin/download/Documentation/Publications/medpage.pdf>
3. <http://eprints.soton.ac.uk/258548/1/ijcai-torsten.pdf>
4. https://vsis-www.informatik.uni-hamburg.de/getDoc.php/publications/289/13_060111-III.4-PaulussenZoellerRothlaufHeinzlBraubachPokahrLamersdorf-FINAL.pdf

- Software

1. <http://jade.tilab.com/>
2. <http://repast.sourceforge.net/>