

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Uso de argumentação em negociação automática

Mauro Miguel Rodrigues



Mestrado Integrado em Engenharia Informática e Computação

Orientador: Ana Paula Rocha

24 de Julho de 2018

Uso de argumentação em negociação automática

Mauro Miguel Rodrigues

Mestrado Integrado em Engenharia Informática e Computação

Resumo

No comércio eletrônico, tal como no comércio tradicional, na negociação dos termos de um negócio, ocorre uma troca de propostas e contrapropostas dos participantes, com o objetivo de obter um acordo aceite por todos. No contexto em análise, que recai sobre a transação de bens entre clientes (compradores) e fornecedores (vendedores), é importante que os acordos beneficiem ambas as partes para que a negociação seja bem sucedida.

O mercado eletrônico deve ser um sistema aberto, o que tem duas implicações imediatas: a informação é extensa (o número de possíveis compradores e vendedores de um mesmo bem pode ser elevado, o número de bens pode ser elevado); e o ambiente é dinâmico (novos participantes podem entrar a qualquer momento e outros podem sair, novos bens podem ser adicionados ao mercado ou removidos do mercado). Estes dois pontos justificam o uso da negociação automática como uma metodologia adequada para a obtenção de um bom acordo. Os participantes do mercado são entidades heterogêneas, com necessidades e perfis negociais diferentes, o que justifica o uso do conceito agente na sua definição.

Protocolos de negociação automática baseada em argumentação são uma alternativa a protocolos de negociação baseados apenas em propostas, uma vez que o uso de argumentação permite que os agentes troquem argumentos que contêm informação mais completa que simples propostas. Os argumentos poderão apelar ou recompensar o oponente, devendo ser usados de acordo com a situação atual do processo negocial. Estes argumentos poderão ainda incluir informação sobre ações passadas ou ainda características contextuais.

A metodologia proposta será validada em cenários simulados no domínio da indústria têxtil e avaliada através de uma análise comparativa de resultados em que a negociação é realizada com e sem recurso à argumentação.

Abstract

In an electronic commerce, similar to the traditional commerce, negotiation happens when participants exchange proposals and counter-proposals with the goal to reach a final agreement. In this context where buyers try to buy products from the sellers it's important that the agreements benefit both parties so that a successful negotiation can be achieved. The electronic market is a open system, where the information available can be extensive (the number of buyers and sellers of the same product can be high), is also a dynamic market (participants can leave at any time and new goods can be added). These two points justify the use of automated negotiation as a methodology to reach a good agreement. The participants of the market are heterogeneous entities with different needs and profiles which justify the use of agents.

Argumentation based negotiation protocols are an alternative to proposal based negotiation protocols, the use of argumentation allow agents to exchange additional and more complex information and not just simple proposals. These arguments may reward or appeal the oponent, and should be use according to each situation. These arguments may included information about past actions and other contextual information.

The use of argumentation in the negotiation will be validated with the comparative analysis of the results obtained in negotiations with and without argumentation in simulated scenarios.

Agradecimentos

Gostaria de agradecer à Professora Ana Paula Rocha por todo o apoio e ajuda prestada durante o desenvolvimento desta dissertação.

À minha namorada, Ília Azevedo, por ter estado sempre ao meu lado, me dar motivação e fazer feliz, mesmo estando num momento muito difícil da sua vida.

Aos meus pais e irmão, que criaram as condições para a minha formação, pelo apoio e crença em mim.

A todos os meus amigos, em especial ao Carlos Alves, Ricardo Duarte e Pedro Silva pelas muitas gargalhadas e companhia ao longo do curso.

Mauro Miguel Rodrigues

*“Be happy for this moment.
This moment is your life.”*

Omar Khayyam

Conteúdo

1	Introdução	1
1.1	Contexto	1
1.2	Motivação e Objetivos	1
1.3	Estrutura do relatório	1
2	Revisão Bibliográfica	3
2.1	Introdução	3
2.2	Sistemas multi-agente	3
2.2.1	Noção de Agente	3
2.2.2	Sistema multi-agente	4
2.2.3	JADE	5
2.3	Negociação Automática	6
2.3.1	Argumentação	8
2.4	Resumo	12
3	Preliminares	13
3.1	Problema	13
3.2	Visão geral da solução	13
3.2.1	Arquitetura do sistema	13
3.2.2	Arquitetura do agente	14
3.2.3	Protocolo de negociação	15
3.3	Resumo	16
4	Implementação: Sistema base sem argumentação	17
4.1	Mercado Eletrónico	17
4.2	Agente Comprador	19
4.2.1	Ciclo de vida do agente comprador	22
4.2.2	Sistema de <i>feedback</i>	22
4.2.3	Cálculo da utilidade	23
4.3	Agente Vendedor	24
4.3.1	Ciclo de vida do agente vendedor	25
4.3.2	Utilidade e formação da proposta inicial	26
4.3.3	Lidar com <i>feedback</i> e cedências	27
4.4	Resumo	28
5	Implementação: Sistema com argumentação	29
5.1	Argumentação	29
5.2	Comunicação entre agentes	31

CONTEÚDO

5.3	Agente Comprador	31
5.3.1	Ciclo de vida do agente comprador	31
5.3.2	Lidar com recompensas e promessas	33
5.4	Agente Vendedor	34
5.4.1	Ciclo de vida do agente vendedor	35
5.4.2	Lidar com apelos	35
5.5	Resumo	36
6	Experimentação e resultados	37
6.1	Cenário 1	37
6.1.1	Negociação sem argumentação	38
6.1.2	Negociação com argumentação	39
6.2	Cenário 2	40
6.2.1	Sistema sem argumentação	40
6.2.2	Sistema com argumentação	41
6.3	Cenário 3	45
6.3.1	Negociação sem argumentação	45
6.3.2	Negociação com argumentação	45
6.4	Cenário 4	49
6.4.1	Sistema sem argumentação	49
6.4.2	Sistema com argumentação	50
6.5	Resumo	54
7	Conclusões e Trabalho Futuro	55
7.1	Contribuições	55
7.2	Trabalho Futuro	55
	Referências	57

Lista de Figuras

2.1	Arquitetura da plataforma JADE [JAD]	6
2.2	Arquitetura de um agente usando negociação clássica [ESM12]	10
2.3	Arquitetura de um agente NBA [ESM12]	11
3.1	Visão geral do sistema	14
3.2	Mecanismo de negociação com argumentação	15
3.3	Mecanismo de negociação sem argumentação	16
4.1	Diagrama de comunicação - Sistema sem argumentação	19
5.1	Diagrama de comunicação - Sistema com argumentação	32
6.1	Cenário 1: Negociação sem argumentação - Utilidade dos vendedores	38
6.2	Cenário 1: Negociação sem argumentação - Utilidade comprador	39
6.3	Cenário 2: Negociação sem argumentação - Utilidade comprador	41
6.4	Cenário 2: Negociação sem argumentação - Utilidade dos vendedores	42
6.5	Cenário 2: Negociação com argumentação - Utilidade comprador	42
6.6	Cenário 2: Negociação com argumentação - Utilidade dos vendedores	43
6.7	Cenário 3: Negociação com argumentação - Utilidade comprador	46
6.8	Cenário 3: Negociação com argumentação - Utilidade dos vendedores	46
6.9	Cenário 4: Negociação sem argumentação - Utilidade comprador	50
6.10	Cenário 4: Negociação sem argumentação - Utilidade dos vendedores	51
6.11	Cenário 4: Negociação com argumentação - Utilidade comprador	52
6.12	Cenário 4: Negociação com argumentação - Utilidade dos vendedores	53

LISTA DE FIGURAS

Lista de Tabelas

4.1	Estrutura de dados da classe <i>BuyerBehaviour</i>	20
4.2	Classe Proposta	20
4.3	Estrutura de dados do Modelo do Agente Comprador	21
4.4	Classe Produto	21
4.5	<i>Feedback</i> dado a cada atributo	24
4.6	Estrutura de dados na classe <i>SellerBehaviour</i>	25
4.7	Estrutura de dados do Modelo do Agente vendedor	25
4.8	Cedência após <i>Feedback</i>	27
5.1	Cedência caso o atributo seja importante	35
5.2	Cedência caso o atributo seja não importante	36

LISTA DE TABELAS

Abreviaturas e Símbolos

ACL	<i>Agent Communication Language</i>
CFP	<i>Call-for-proposal</i>
DF	<i>Directory Facilitator</i>
FIPA	<i>Foundation of Intelligent Physical Agents</i>
JADE	<i>Java Agent Development Environment</i>
NBA	Negociação baseada em Argumentação
SMA	Sistema multi-agente

Capítulo 1

Introdução

1.1 Contexto

No comércio eletrônico, os participantes têm como objetivo a obtenção de acordos para a transação de bens ou produtos, que são geralmente obtidos pela troca de propostas e contrapropostas entre as duas partes do processo: cliente e fornecedor. Os participantes são agentes computacionais, que representam entidades heterogêneas com perfis e atitudes negociais diferentes, e tentam maximizar os seus ganhos no acordo. Para a negociação ser bem sucedida é importante garantir que os acordos finais beneficiam ambas as partes.

1.2 Motivação e Objetivos

A negociação baseada em argumentação tem vindo a ter especial atenção em sistemas multi-agente pois proporciona uma forma natural de resolver conflitos de interesses. No entanto, grande parte dos mecanismos de negociação com argumentação atualmente existentes são trabalhos com uma forte componente teórica, fornecendo apenas *frameworks* formais algo distantes de aplicações da vida real [ZJH10].

Nesta dissertação será desenvolvido um sistema de negociação automática num contexto de comércio eletrônico onde os participantes da negociação (agentes) poderão trocar mais informação do que simples propostas possibilitando assim que utilizem as suas crenças e atitudes como argumentos para se justificarem ou tentarem influenciar o comportamento dos seus oponentes.

1.3 Estrutura do relatório

O capítulo 2 contém a revisão bibliográfica nos tópicos relacionados com a presente dissertação, nomeadamente sistemas multi-agente e negociação automática, focando-se principalmente na negociação automática baseada em argumentação.

Introdução

O capítulo 3 apresenta uma visão geral do problema e uma versão preliminar da solução a ser implementada.

De seguida são descritas as implementações do sistema base sem argumentação, capítulo 4, e do sistema com argumentação, capítulo 5.

O capítulo 6 define os cenários de testes e apresenta os resultados.

O capítulo 7 indica as contribuições e as possibilidades de trabalho futuro.

Capítulo 2

Revisão Bibliográfica

2.1 Introdução

Este capítulo descreve o estado da arte relativo aos tópicos usados na presente dissertação e apresenta trabalhos existentes relacionados com o problema enunciado. Serão analisados com especial atenção os tópicos de sistemas multi-agente e negociação automática.

2.2 Sistemas multi-agente

2.2.1 Noção de Agente

Agentes autónomos são sistemas computacionais que habitam num ambiente dinâmico e complexo, sentem e reagem autonomamente neste ambiente de modo a alcançar um objetivo ou realizar uma tarefa para a qual foram projetados [Mae95].

Segundo o autor [Mae95], os agentes autónomos podem assumir várias formas de acordo com o ambiente onde estão inseridos, se o ambiente for real e físico o agente poderá tomar a forma de um robô autónomo, da mesma maneira que se podem criar agentes virtuais para ambientes de simulação e digitais. Dependendo do tipo de ambiente onde são inseridos, os agentes poderão também assumir diferentes tipos de comportamentos.

Russel [RN10] enumera as seguintes características para definir o ambiente:

- Totalmente observável / Parcialmente observável - Se a cada momento os sensores do agente lhe derem acesso ao estado completo do ambiente, diz-se que o ambiente é totalmente observável caso contrário diz-se que o ambiente é parcialmente observável. A grande vantagem de ser totalmente observável é não ser necessário guardar qualquer informação para manter informação sobre o mundo.

Revisão Bibliográfica

- Determinístico / Estocástico - Se o próximo estado do ambiente é completamente determinado pelo estado atual e pela ação do agente diz-se que o ambiente é determinístico, caso isto não se verifique diz-se que o ambiente é estocástico.
- Episódico / Sequencial - Num ambiente episódico o comportamento do agente é dividido em episódios atômicos onde a cada episódio o agente tem uma percepção e reage fazendo uma única ação, sendo que o próximo episódio não depende das ações tomadas anteriormente. Num ambiente sequencial uma ação poderá influenciar todas as decisões futuras.
- Estático / Dinâmico - Se o ambiente poder ser alterado enquanto o agente está a deliberar diz-se que o ambiente é dinâmico, caso contrário é estático.
- Discreto / Contínuo - Esta distinção aplica-se ao estado do ambiente, à maneira de como o tempo é tratado, e às percepções e ações do agente. Utilizando o exemplo do autor, um jogo de xadrez tem um número finito de estados, e um conjunto discreto de percepções e ações enquanto que, por exemplo, conduzir um táxi é um estado contínuo e um problema contínuo de tempo, a localização do táxi e dos outros veículos encontram-se num intervalo contínuo.
- Conhecido / Desconhecido - Diz-se que o ambiente é conhecido se o agente tiver conhecimento sobre como funciona esse ambiente. Num ambiente contínuo o resultado de uma ação é já conhecida pelo agente, enquanto que num ambiente desconhecido este resultado não é conhecido e o agente precisa de aprender para tomar as melhores decisões.

2.2.2 Sistema multi-agente

Um sistema multi-agente (SMA) é constituído por um conjunto de agentes que, em conjunto, fazendo uso das suas competências globais, cooperam com a finalidade de atingirem um objetivo comum. A utilização de técnicas de cooperação eficientes permite ao sistema apresentar uma maior capacidade do que a soma das capacidades de cada agente individualmente [Nog02]. A cooperação entre os diferentes agentes permite uma resolução de problemas mais eficiente. Para a interação ser possível, é necessário que os agentes partilhem conhecimento e que entendam esse conhecimento, sendo por isso requeridos as seguintes competências a um sistema multi-agente [Nog02]:

- Plataforma de comunicação. A plataforma de comunicação disponibiliza o meio físico de transmissão de informação entre os diversos agentes que constituem o SMA. Na secção 2.2.3 apresenta-se uma descrição da plataforma JADE, por se tratar de uma plataforma aceite e usada por muitos investigadores da área.
- Linguagem de comunicação. Em sistemas multi-agentes os agentes poderão trocar desde informações simples até questionar e requerer informações de outros agentes. A linguagem ACL (*Agent Communication Language*) é uma linguagem adotada pela FIPA (*Foundation of Intelligent Physical Agents*) e que permite a comunicação quer de conteúdos quer de atitudes relativas ao conteúdo da mensagem que é trocada [FP97].

- Ontologia. A ontologia trata-se de uma especificação de um vocabulário para um domínio partilhado [Gru93]. Em SMA é necessário assegurar que existe um entendimento da informação correto por parte de todos os agentes sendo fundamental uma ontologia comum. A ontologia define classes, relações, conceitos e outros objetos de um domínio [Gru93]. O conceito de ontologia não será objeto de estudo nesta dissertação, pelo que será assumida a existência de uma ontologia comum, garantindo assim a consistência e compatibilidade da informação.

2.2.3 JADE

JADE é uma *framework* que simplifica o desenvolvimento e implementação de sistemas multi-agente em conformidade com as especificações FIPA¹ (*Foundation for Intelligent Physical Agents*). JADE oferece as seguintes *features* ao desenvolvedor [BPR99]:

- Plataforma de agentes compatível com as especificações FIPA.
- Plataforma para agentes distribuídos.
- Interface para simplificar o registo de serviços de agentes com um ou mais domínios.
- Mecanismo de transporte e interface para enviar/receber mensagens de/para outros agentes.
- Transporte *light-weight* de mensagens ACL dentro da mesma plataforma do agente.
- Biblioteca com protocolos FIPA prontos a serem usados.
- Registo automático de agentes através da AMS (*Agent Management System*).
- Nomes compatíveis com especificações FIPA, sempre que são iniciados os agentes recebem um identificador único global.
- Interface gráfica para gerir vários agentes e plataformas de agentes.

O JADE simplifica a implementação de sistemas multi agente, quer pela inclusão de um *middleware* que está em conformidade com as especificações FIPA, quer pela inclusão de um conjunto de ferramentas gráficas que suportam essencialmente a fase de depuração do sistema. A arquitetura da plataforma JADE é representada na figura 2.1.

A plataforma é formada por contentores (*containers*), onde residem os vários agentes. O processo de inicialização e gestão da vida de um agente, e a sua inclusão num contentor, é da responsabilidade do JADE, que disponibiliza todos os serviços necessários. Existe um contentor especial, denominado *Main Container*, onde residem obrigatoriamente os agentes de serviços da plataforma (AMS, DF), que deve ser inicializado em primeiro lugar e estabelece o ponto de início de execução da plataforma.

¹FIPA é uma organização sem fins lucrativos fundada em 1995. Tem como objetivo criar padrões de desenvolvimento de sistemas multi-agentes e de interoperabilidade de agentes heterogéneos e interativos

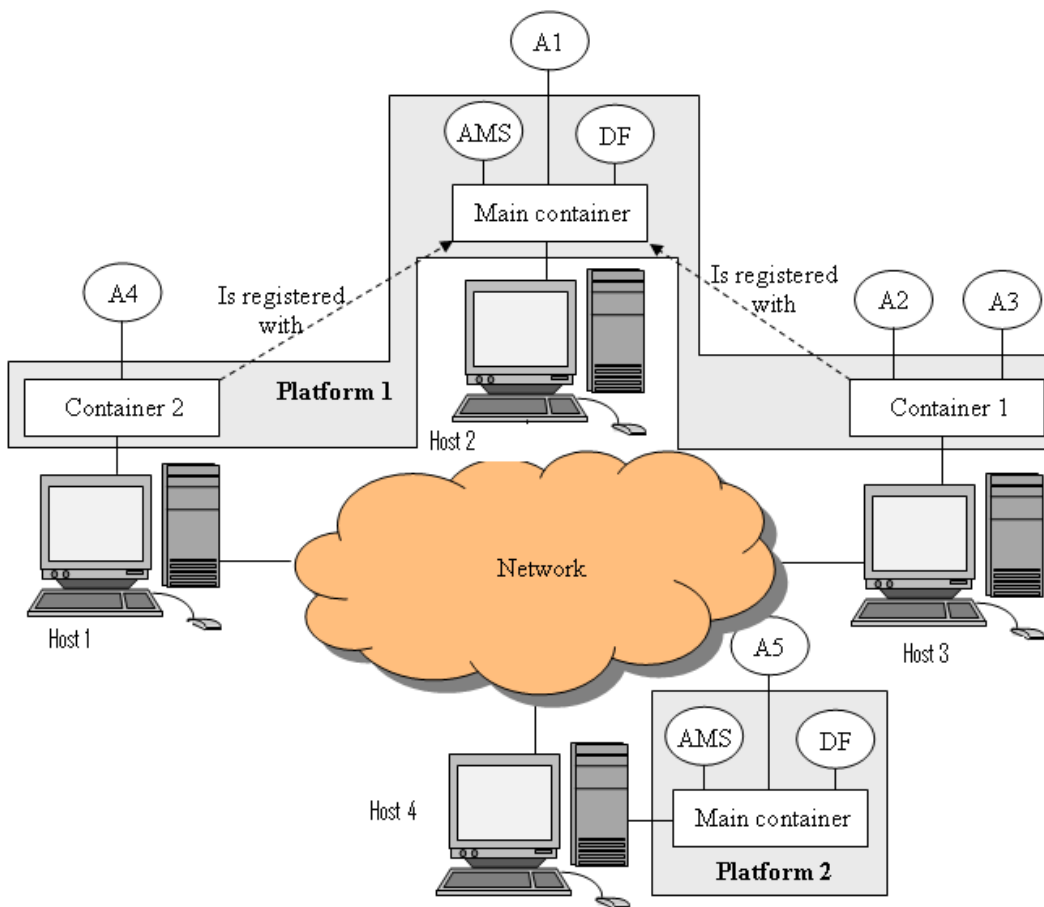


Figura 2.1: Arquitetura da plataforma JADE [JAD]

JADE utiliza também a abstração *Behaviour* para modelar as tarefas que o agente é capaz de efetuar sendo que cada agente instancia os seus *behaviours* de acordo com as suas necessidades e capacidades [BPR99]. A *framework* fornece três tipos de *behaviours* primitivos:

- *SimpleBehaviour* - *Behaviour* que deverá ser usado para implementar ações atômicas do trabalho do agente.
- *CyclicBehaviour* - *Behaviour* que executa para sempre.
- *OneShotBehaviour* - *Behaviour* que apenas executam uma vez.

2.3 Negociação Automática

Segundo Jennings [JPNS], as interações são intrínsecas a todos os sistemas multi-agente, e manifestam-se em três cenários: comunicação (facilitar a troca de informação), coordenação (organização de atividades individuais de maneira coerente) e colaboração (agentes trabalham em

conjunto para um objetivo comum). O principal mecanismo para gerir estas interações é a negociação que consiste na troca de informação entre agentes de modo a conseguirem atingir um acordo. A negociação pode ser usada em diferentes contextos, mas abrange normalmente três tópicos: protocolos de negociação, objetos de negociação e modelos de decisão dos agentes.

- Protocolos de negociação: Conjunto de regras que dirigem a negociação e inclui o tipo de participantes permitido, os estados de negociação (por exemplo: negociação fechada ou a aceitar propostas), os eventos que alteram o estado da negociação (por exemplo: sem participantes ou proposta aceite) e as ações permitidas aos participantes nos diferentes estados de negociação.
- Objetos de negociação: Conjunto de tópicos sobre os quais é necessário chegar a acordo. Podem incluir desde algo simples como o preço de um produto até algo mais complexo como preços, qualidade ou datas de entrega. Inclui também as operações que podem ser realizadas sobre estes tópicos, por exemplo no caso de os tópicos serem não negociáveis os participantes apenas podem ter uma atitude de *take it or leave it* não havendo espaço para qualquer tipo de contra-proposta. Por outro lado pode ser permitido apresentar contra-propostas de modo a tentar que os termos da negociação mais favoreçam o participante.
- Modelos de decisão dos agentes: O conjuntos de ações que o agente emprega de modo a chegar aos seus objetivos de negociação

Ainda segundo o autor, a importância destes três tópicos é relativa e varia entre negociações e contextos, em alguns casos o protocolo de negociação é dominante, noutros casos o modelo de decisão é mais importante, não existindo assim uma técnica de negociação universal que seja melhor que as outras, dependendo do tipo de situação que se enfrenta. Os mecanismos de negociação automática podem ser agrupados em quatro grupos [JPNS] [OR01]:

- Leilões - Este tipo de negociação é muito popular devido à sua simplicidade e regras pré-definidas. O leilão é um processo de compra e venda de produtos onde um leiloeiro recebe ofertas dos participantes sobre produto, vendendo o item ao participante com a oferta mais alta. Podem no entanto haver várias variações nos tipos de leilões como por exemplo o leilão Inglês (onde os preços das ofertas vão aumentando) e o leilão Holandês (onde os preços das ofertas vão diminuindo) [ZL10].
- Teoria de jogo - Os mecanismos de teoria de jogo apenas podem ser utilizados em cenários onde exista informação e racionalidade perfeita.
- Negociação baseada em heurísticas - As propostas geradas pelos participantes são resultantes de uma combinação linear de um conjuntos de táticas de negociação. Permite uma negociação mais flexível que as anteriores.
- Negociação baseada em argumentação - O uso de argumentos permite formas de interação mais sofisticadas que as anteriores. Este mecanismo de negociação será o usado no presente trabalho e por isso é descrito com mais detalhe na secção seguinte.

2.3.1 Argumentação

No seu trabalho sobre negociação Jennings [JPNS] concentra-se na negociação baseada em argumentação. Neste tipo de negociação os agentes trocam não apenas propostas mas também informações adicionais que têm como objetivo não apenas justificar a decisão do agente mas também tentar persuadir o oponente a alterar o seu estado interno.

De modo a que a negociação seja eficiente o agente destinatário tem de possuir um mecanismo para dar *feedback*, para que o agente com o qual está a negociar saiba o porquê de ter rejeitado uma dada proposta e não apenas que a proposta foi rejeitada. Este mecanismo de *feedback* pode tomar a forma de críticas ou contra-propostas [JPNS] :

- Crítica: Uma crítica proporciona duas formas de *feedback*, permite introduzir restrições a um atributo particular da negociação e permite indicar a aceitação/rejeição de partes da proposta.

A: Eu proponho que me forneças o serviço X segundo as seguintes condições.

B: Concordo com o preço de X mas a data de entrega é demasiado tarde.

A: Eu proponho fornecer-te o serviço Y se me forneceres o serviço X

B: Eu não quero o serviço Y.

No primeiro exemplo a crítica indica que partes da proposta aceita e introduz restrições num dos atributos (a data de entrega tem de ser menor do que a atual). No segundo exemplo a crítica rejeita parte da proposta.

- Contra-proposta: Uma contra-proposta é simplesmente uma proposta mais favorável ao agente em resposta a uma proposta anterior.

A: Eu proponho fornecer-te o serviço Y se me forneceres o serviço X.

B: Eu proponho fornecer-te o serviço X se me forneceres o serviço Z.

Neste exemplo o agente B gera uma proposta mais favorável para si, emendando a proposta inicial.

Por si só as críticas e contra-propostas são afirmações do que os agentes pretendem e não permitem justificar a posição do agente na negociação ou persuadir o oponente a mudar a sua posição pelo que é necessário a introdução de informações adicionais na forma de argumentos. As categorias de argumentos mais comuns são: ameaças, recompensas, apelos [JPNS] .

- Ameaças: É o tipo de argumento mais agressivo, consiste em avisar o oponente que se rejeitar a proposta algo negativo lhe irá acontecer. Por exemplo, caso o oponente rejeite a proposta deixará de fazer parte das próximas rondas negociais.

Revisão Bibliográfica

- Recompensas: Neste tipo de argumento o agente propõe que se o oponente aceitar a proposta algo bom lhe irá acontecer. Por exemplo, caso o oponente aceite a proposta o agente promete que facilitará novas negociações no futuro.
- Apelos: É o tipo de argumento menos agressivo, consiste em apelar ao oponente que aceite a proposta por algum motivo.

Kraus et al. [KSE98] usa também esta categorização de argumentos, mas com uma decomposição mais granular e classifica tais argumentos, por ordem crescente de agressividade, como:

- Apelo a práticas existentes - O agente J recebe uma rejeição por parte do agente I para realizar a ação C na medida que vai contra o objetivo O de I. No entanto J acredita que outro agente H fez a mesma ação C e não foi contra o mesmo objetivo O de H, usando-a como argumento.
- Contra-exemplo - Neste caso o agente J pede ao agente I que faça uma ação C, o agente I recusa. O agente J acredita que a razão de I ter recusado é que a ação vai contra os seus objetivos, mas J crê que no passado I tenha feito uma ação B que também fosse contra os seus objetivos, usando-a como contra-exemplo.
- Apelo para promessas passadas - Neste caso o agente J pede ao agente I para fazer uma ação C devido a uma promessa passada, caso o agente recuse, J relembra o agente I sobre a promessa.
- Apelo ao interesse próprio - O agente J acredita que a ação C é a favor dos objetivos do agente I, por isso usa-o como argumento, este argumento é útil quando J crê que I não está ciente desta implicação.
- Promessa de recompensas futuras - O agente J atrai o agente I para fazer a ação C, oferecendo-se para fazer a ação D no futuro.
- Ameaças - O agente J pede ao agente I para fazer a ação C, ameaçando que caso não faça C, J fará a ação A.

Para modelar e desenvolver agentes capazes de negociarem usando argumentação, doravante designados por agentes NBA (Negociação baseada em argumentação) é necessária a inclusão de funcionalidades adicionais capazes de tratar este novo paradigma [JPNS]:

- Mecanismos para trocar propostas e argumentos de modo a que o oponente as entenda.
- Técnicas para gerar propostas e os respetivos argumentos - Para gerar propostas é necessário duas atividades: instanciar o objeto de negociação de acordo com a função de avaliação do agente e determinar que argumentos deverão acompanhar a proposta para maximizar a probabilidade de ser aceite, a complexidade da determinação de argumentos pode variar desde uma pequena lógica a manter um modelo complexo do oponente e tentar fazer previsões deste modelo.

Revisão Bibliográfica

- Técnicas para avaliar propostas e os respetivos argumentos - De modo a saber como responder o agente precisa de avaliar as propostas recebidas, esta avaliação é dividida em duas fases: avaliar o objeto de negociação e avaliar o impacto dos argumentos respetivos. Disto podem resultar diferentes ações: o objeto de negociação pode ser aceite tal como está; o objeto de negociação não é aceite mas os argumentos de suporte permitem que a proposta seja aceite; o objeto de negociação não é aceite e os argumentos fornecidos também não permitem que a proposta seja aceite.
- Técnicas para responder a propostas e os respetivos argumentos - Após avaliar a proposta o agente precisa de responder, podendo aceita-la, rejeita-la, gerar uma crítica ou uma contra-proposta. No caso de uma crítica o agente deverá determinar que componentes do objeto de negociação aceita e que componentes rejeita, tem de decidir também que argumentos deverá incluir para dar suporte à crítica (pode optar por não usar argumentos) e como responder aos argumentos que acompanharam a proposta, podendo ignorá-los ou tentar enfraquece-los. No caso de uma contra-proposta o procedimento é semelhante no entanto em vez de dar *feedback* o agente deverá instanciar o objeto de negociação com os valores desejados.

Nas figuras seguintes encontram-se duas arquiteturas de agentes de negociação, a primeira sendo a de um agente usando negociação clássica (Figura 2.2) e a segunda de um agente NBA (Figura 2.3).

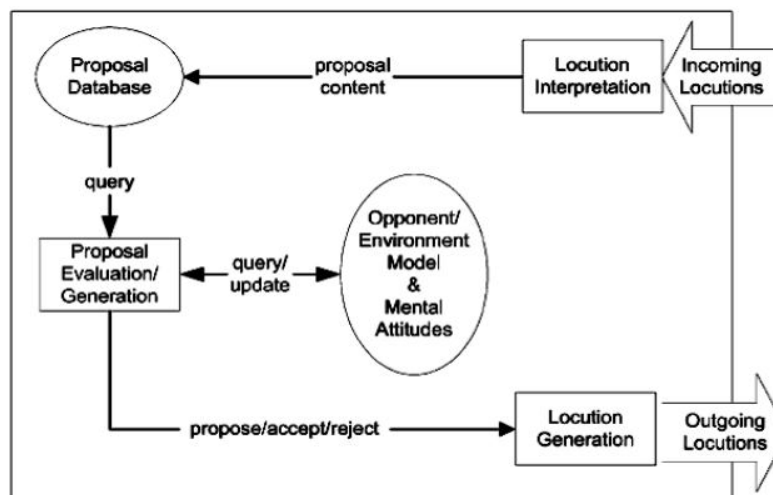


Figura 2.2: Arquitetura de um agente usando negociação clássica [ESM12]

Na arquitetura de um agente usando negociação clássica (Figura 2.2), as locuções contendo as propostas são interpretadas e posteriormente avaliadas tendo como referências na avaliação o oponente, o modelo do ambiente e as atitudes do agente [ARL03]. Estas referências são também atualizadas à medida que as propostas são avaliadas. Após avaliação da proposta, é gerada uma locução contendo o resultado (proposta/aceitação/rejeição) e enviada para o oponente.

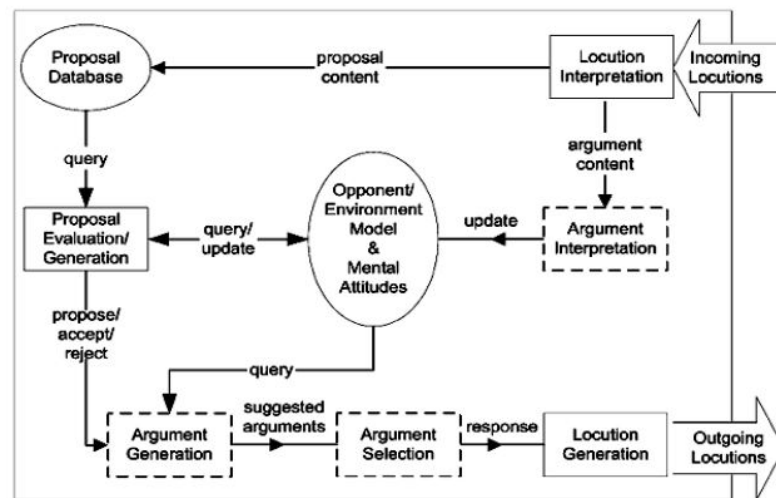


Figura 2.3: Arquitetura de um agente NBA [ESM12]

Num agente de negociação NBA (Figura 2.3) existem três módulos adicionais, interpretação de argumentos, geração de argumentos e seleção de argumentos [ARL03]. O módulo de interpretação de argumentos tem como principal foco perceber o conteúdo do argumento, atualizar o estado interno do agente de acordo com esse argumento e possivelmente mudar as suas atitudes. Tal como num agente clássico as propostas são interpretadas e avaliadas, no entanto existe o componente de geração de argumentos que permite a criação de argumentos, que podem ser de suporte a uma proposta ou um apenas um pedaço adicional de meta-informação. Posteriormente caso existam vários argumentos que possam ser usados será utilizado a componente de seleção de argumentos para escolher o mais apropriado para a situação

Argumentação em negociação multi-atributo

Dong et al. [DJH⁺13] propõem um modelo de negociação multi-atributo baseada em argumentação de fatores internos. Neste modelo um agente comprador e um vendedor tentam chegar a acordo sobre um produto que é caracterizado por ter vários atributos, sendo que cada atributo tem um peso diferente na função de utilidade de cada agente. Os fatores internos como a utilidade, preferência de atributos ou objetivos de negociação são trocados entre agentes na forma de argumentos, esta argumentação pode facilitar a cedência do oponente e permitir alcançar uma proposta que beneficie os dois agentes. Os autores referem alguns tipos de conflitos que podem acontecer durante a negociação e como resolve-los:

- Conflito 1: Quando um agente recebe uma proposta em que um dos valores dos atributos está fora do seu intervalo.
- Conflito 2: Quando um agente recebe uma proposta em que a utilidade é menor do que a utilidade mínima.

Revisão Bibliográfica

- Conflito 3: Quando um agente recebe uma proposta em que a utilidade é maior do que a utilidade mínima mas é menor do que a utilidade da oferta proposta por este na próxima ronda.

De modo a resolver estes conflitos desenvolveram o seguinte mecanismo de negociação com argumentação:

- Caso aconteça o conflito 1, por exemplo, ao receber uma proposta em que o valor do atributo x é 200 mas o máximo do intervalo é 150, o agente deverá atacar argumentando que o valor ultrapassa 150. Caso o oponente não ceda, a negociação termina.
- Caso aconteça o conflito 2 ou 3, o agente deverá enviar informações sobre qual é o atributo mais importante para si naquela ronda negocial, esperando obter uma cedência por parte do oponente no valor daquele atributo.

Este tipo de estratégia de negociação, num sistema não competitivo, baseia-se em tentar aumentar a utilidade do oponente sem diminuir a sua, informando-o sobre as suas preferências para que possa ceder em alguns atributos menos importantes para si mas com um peso superior para o seu oponente.

2.4 Resumo

Neste capítulo foi apresentada a revisão bibliográfica onde foram descritos e discutidos os conceitos relevantes para a dissertação, sendo apresentados os conceitos de agente e sistema multi-agente, a ferramenta a ser utilizada no desenvolvimento do trabalho (JADE). Foi também abordado o tema de negociação automática onde o principal foco foi a negociação baseada em argumentação.

Capítulo 3

Preliminares

3.1 Problema

Numa negociação automática baseada em propostas os participantes interagem pela troca iterativa de valores relativos aos atributos da transação em negociação até chegarem a um acordo. Neste contexto de comércio eletrônico, baseado em *Business-to-business*, a negociação envolve a venda de produtos e implica análises cuidadosas por parte dos agentes que pretendem obter o melhor acordo possível para si, tendo em conta as suas funções de utilidade. No entanto as negociações baseadas em propostas podem ser limitativas, pois apenas é trocada informação sobre os valores das propostas e não qualquer justificação ou razão para aceitar determinada proposta, o que poderá levar a negociações longas e que não alcancem os objetivos esperados [FRN09].

3.2 Visão geral da solução

A criação de uma camada de argumentação no topo da negociação automática visa enriquecer as trocas de propostas com informação justificativa ou de suporte a essas mesmas propostas, tentando assim melhorar o processo negocial.

De modo a avaliar o potencial do uso de argumentação na negociação, serão desenvolvidos dois sistemas e efetuada a comparação sobre os resultados obtidos em cada um: um sistema base sem argumentação e um sistema com argumentação.

3.2.1 Arquitetura do sistema

O sistema a considerar é povoado por agentes compradores e agentes vendedores. Os agentes compradores procuram adquirir um produto e os agentes vendedores procuram vender produtos. O produto é descrito por um conjunto de atributos que serão sujeitos a negociação, sendo, no contexto do presente trabalho, considerados os atributos preço, quantidade e data de entrega. Considera-se que a oportunidade de mercado se refere a uma necessidade de compra, isto é, existe um único agente comprador que pretende adquirir um produto específico e possíveis vários agentes vendedores que estarão disponíveis para vender esse produto. (figura 3.1)

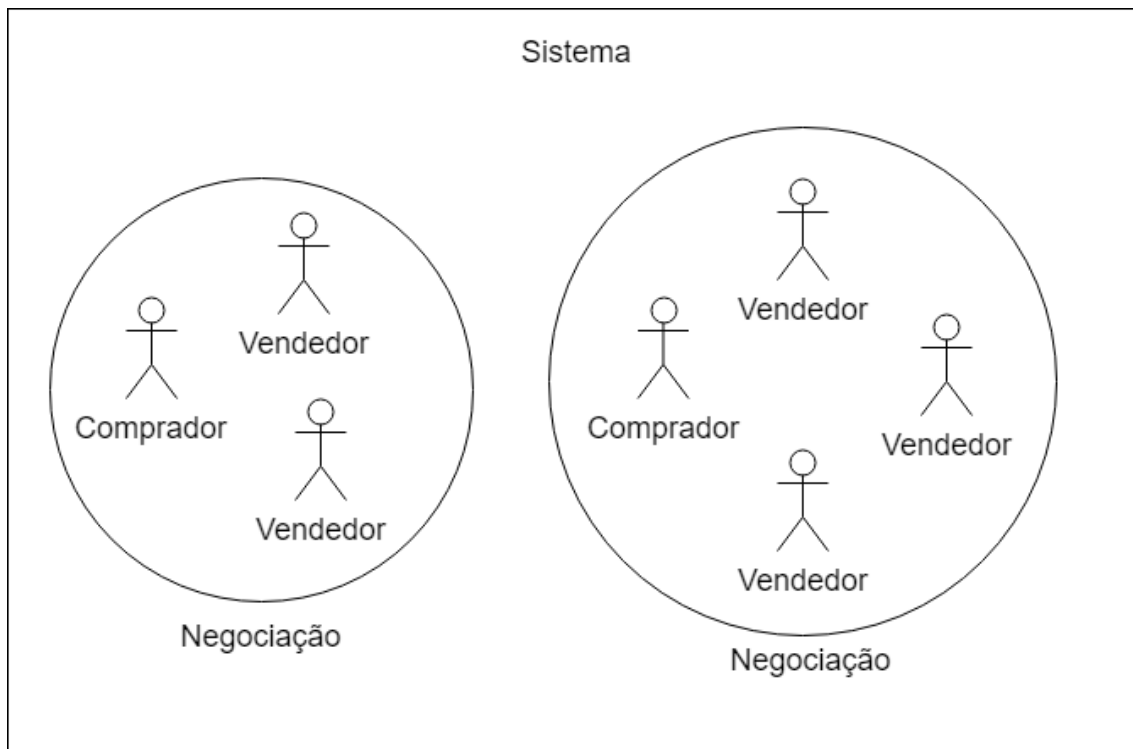


Figura 3.1: Visão geral do sistema

O agente comprador inicia a negociação com um pedido de compra e, posteriormente, os agentes vendedores que tenham o produto disponível entram na negociação. Trata-se pois de uma negociação, em termos de número de participantes, identificada como sendo de um-para-muitos.

No sistema sem argumentação, os agentes vendedores enviam propostas em resposta ao pedido de compra, que serão sujeitas ao *feedback* do agente comprador.

No sistema com argumentação os agentes trocam não apenas propostas, mas acoplada a cada proposta pode ser enviado um argumento. Este argumento será composto por dois campos: uma pretensão, que neste caso será a proposta, e uma razão, que será a justificação apresentada para convencer o oponente a aceitar a proposta.

É apresentado na figura 3.3, de forma resumida, o mecanismo de negociação para o sistema sem argumentação, e na figura 3.2 o mecanismo proposto para o sistema com argumentação. Ambos os processos serão apresentados com maior detalhe nos capítulos 5 e 4, respetivamente.

3.2.2 Arquitetura do agente

A arquitetura de um agente comprador ou vendedor inclui três módulos principais:

- Módulo de interpretação
- Módulo de decisão
- Módulo de geração de locuções (propostas/respostas)

Preliminares

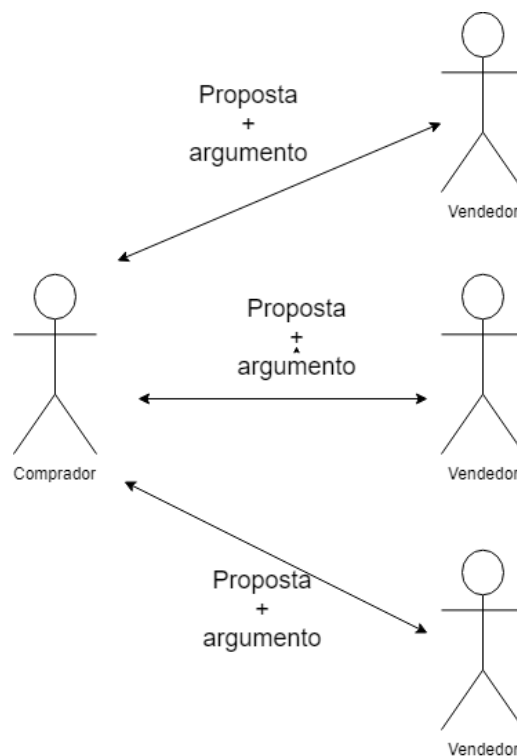


Figura 3.2: Mecanismo de negociação com argumentação

O módulo de **interpretação** é responsável por decifrar e entender as mensagens recebidas de outros agentes. O módulo de **geração de locuções** será responsável por criar a mensagem com a proposta/resposta do agente e enviar para os oponentes. Estes dois módulos servirão meramente para comunicação com os outros agentes. O módulo de **decisão** é responsável por toda a inteligência do agente e é onde são tomadas todas as decisões do agente. Neste módulo estão incluídas as preferências do agente, as suas funções de utilidade, objetivos de negociação e atitude.

3.2.3 Protocolo de negociação

No sistema com argumentação, o agente comprador envia pedidos de compra com informação sobre o produto que pretende (identificação do produto e domínios da negociação) a todos os agentes do mercado que possuam esse produto para venda. Os agentes vendedores respondem com uma proposta a este pedido, caso tenham o produto disponível e o domínio da negociação lhes interesse. Caso contrário, rejeitam a negociação. Após a proposta inicial inicia-se um ciclo de envio de propostas e argumentos por parte de ambos os agentes. Este processo é iterativo e repete-se até que o agente comprador aceite uma proposta ou o vendedor não esteja disposto a ceder mais.

No sistema sem argumentação, o agente vendedor também envia pedidos de compra com informação do produto que pretende (identificação e domínios da negociação), sendo que os agentes vendedores respondem apenas com propostas. Posteriormente segue-se um ciclo de *feedback* às

Preliminares

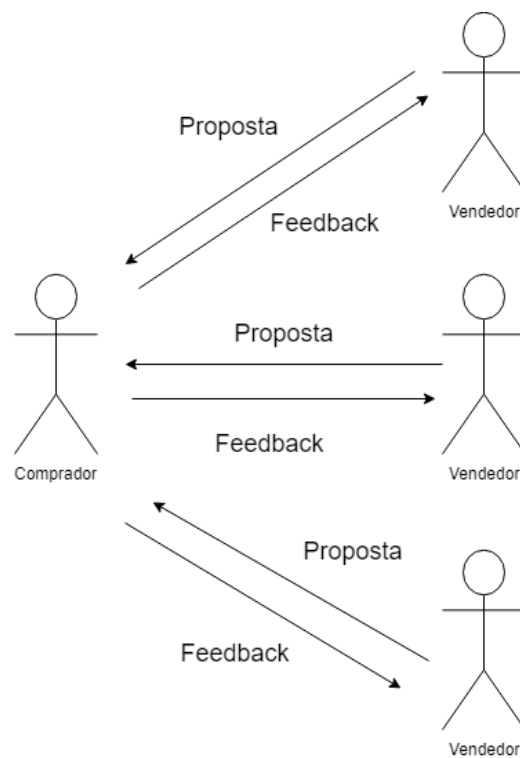


Figura 3.3: Mecanismo de negociação sem argumentação

propostas por parte do agente comprador que leva a uma contra-proposta por parte do agente vendedor. Este processo repete-se até que o agente comprador aceite uma proposta ou o agente vendedor não esteja disposto a ceder mais.

3.3 Resumo

Neste capítulo foi descrito o problema e foi apresentado uma visão geral da proposta de solução. Foram apresentados os componentes do sistema, que serão descritos com maior detalhe nos próximos capítulos.

Capítulo 4

Implementação: Sistema base sem argumentação

4.1 Mercado Eletrónico

O mercado comporta a existência de dois tipos de agentes, agentes compradores e agentes vendedores, que se comportam como oponentes no processo de negociação. Assim, estes agentes apresentam necessariamente comportamentos diferentes, que são descritos nas secções 4.2 e 4.3, respetivamente. Outros componentes que interessa caracterizar num mercado eletrónico são identificados nos parágrafos seguintes.

Participantes no mercado

Como já foi referido, neste mercado a negociação é identificada como uma negociação "um-para-muitos", pois é realizada entre:

- um comprador: agente que inicia a negociação, indicando ao mercado a necessidade de obtenção de um produto
- vários vendedores: agentes que possuem o produto pretendido pelo comprador, disponível para venda

O mercado é implementado fazendo uso da plataforma JADE, que, como já foi referido, possui ferramentas próprias para a gestão de sistemas multi agente. Todos os agentes presentes no mercado se registam na plataforma JADE, dando a conhecer ao mercado a sua identidade e localização e ainda os seus serviços. Neste mercado, os serviços são: para os agentes compradores, os produtos que estes estão interessados em comprar; para os agentes vendedores, os produtos que estes possuem para venda. Estes serviços são guardados no módulo *Directory Facilitator* do JADE, e permite pesquisas posteriores por parte de agentes interessados num serviço específico.

Objeto de negociação

O produto solicitado pelo agente comprador constitui o objeto de negociação e é caracterizado pelos seguintes quatro atributos:

- Nome do produto
- Preço
- Data de entrega
- Quantidade

Os valores dos atributos preço, data de entrega e quantidade são negociáveis, isto é, são discutidos e alterados pelos agentes intervenientes na negociação enquanto esta decorrer. Cada agente possui um intervalo de valores pré-definido para estes atributos, determinado de acordo com as suas preferências (comprador) ou competências (vendedor). É este intervalo de valores que o agente pode negociar.

Comunicação entre agentes

A comunicação entre os agentes presentes no mercado é essencial para que a negociação ocorra. É necessário definir quais os tipos de mensagens que os agentes podem trocar e respectivos conteúdos. Para tal, foi usado o *standard* ACL (*Agent Communication Language*), definido pela FIPA no âmbito do uso de sistemas multi agente. Este *standard* é também suportado pela plataforma JADE, que foi a plataforma usada no desenvolvimento do trabalho. Em ACL, uma mensagem é caracterizada por múltiplos campos, sendo os mais relevantes a *performative*, que identifica o tipo ou sentido semântico da mensagem ACL, e o conteúdo.

Os tipos de mensagens (*performative*) incluídas no sistema desenvolvido nesta dissertação são:

- **CALL FOR PROPOSAL (CFP)** - É a mensagem iniciadora da negociação, só pode ser utilizada pelo agente comprador e tem como conteúdo o nome do produto que o agente procura comprar assim como os limites máximos e mínimos para os seus atributos (domínio de negociação).
- **PROPOSE** - Esta mensagem é apenas utilizada pelos agentes vendedores na resposta ao pedido de compra ou feedback do comprador. Tem como conteúdo a proposta que o agente faz, que inclui três valores: preço, número de dias de entrega e quantidade.
- **REFUSE** - Esta mensagem é utilizada pelos agentes vendedores, tem conteúdo vazio e serve para recusar ou abandonar a negociação.
- **INFORM** - Esta mensagem é enviada apenas pelo agente comprador e tem como conteúdo o *feedback* (ver 4.2.2) dado a uma proposta enviada pelo vendedor.

- **ACCEPT_PROPOSAL** - Esta mensagem é utilizada pelo agente comprador aquando do fim da negociação para avisar o agente vendedor que a sua proposta foi aceite.
- **REJECT_PROPOSAL** - Esta mensagem é enviada pelo agente comprador no final da negociação para todos os agentes vendedores que não apresentaram a melhor proposta.

Na figura 4.1 encontra-se o diagrama de comunicação relativo ao processo de negociação entre agente comprador e agentes vendedores, no sistema sem argumentação.

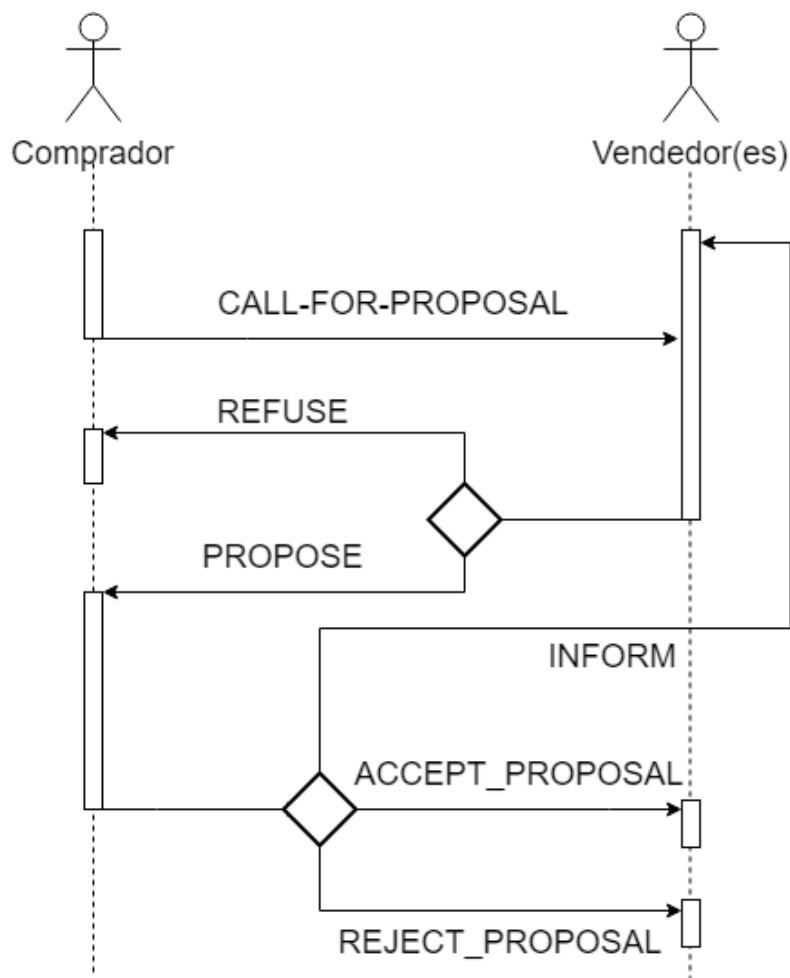


Figura 4.1: Diagrama de comunicação - Sistema sem argumentação

4.2 Agente Comprador

O agente comprador é o agente que pretende adquirir um produto específico do mercado. É responsável por pesquisar no mercado os agentes vendedores que têm registado os produtos que lhe interessam, após o que inicia e regula a negociação.

Implementação: Sistema base sem argumentação

Para implementar o agente comprador foram desenvolvidas duas classes responsáveis pelos comportamentos que este agente exhibe, *BuyerBehaviour* (tabela 4.1) e *BuyerDecisionModel* (tabela 4.3).

A classe *BuyerBehaviour* estende a classe *Behaviour* da *framework* JADE e permite implementar o comportamento geral do agente comprador.

A classe *BuyerDecisionModel* implementa as funções específicas de cálculo de utilidades e funções de tomada de decisão relativos ao processo de negociação.

Nas tabelas abaixo estão enumeradas as estruturas presentes na classe *BuyerBehaviour*.

Atributos	Estrutura de dados
Modelo	<i>BuyerDecisionModel</i> (tabela 4.3)
proposals	HashMap<IDSeller, Proposta>
round	Integer
bestProposal	Proposta (tabela 4.2)
bestUtility	Double
sellers	Array<DFAgentDescription>
averageUtility	Double
averageUtilityLasRound	Double

Tabela 4.1: Estrutura de dados da classe *BuyerBehaviour*

Atributos	Estrutura de dados
message	ACLMessage
price	Double
quantity	Double
delivery	Double
utility	Double

Tabela 4.2: Classe Proposta

Como é possível ver pela tabela 4.1, a classe *BuyerBehaviour* é constituída pelas seguintes estruturas de dados:

- Um modelo do agente, que é descrito em mais detalhe na tabela 4.3;
- um *HashMap* de propostas, esta estrutura guarda todas as propostas recebidas; foi escolhida a estrutura *hashmap*, pois permite o acesso rápido à proposta de um vendedor dando como *key* o seu ID;
- um inteiro que representa a ronda atual;
- a melhor proposta até ao momento, de relembrar que foi criada uma estrutura Proposta (tabela 4.2) para guardar mais convenientemente a informação de uma proposta;

Implementação: Sistema base sem argumentação

- um *double* que tem a melhor utilidade passada;
- um *array* de vendedores ativos, este *array* é construído após procura no serviço DFS de vendedores (ver 4.2.1);
- a utilidade média na ronda atual;
- a utilidade média na ronda passada;

O modelo do agente (tabela 4.3) inclui as suas preferências e valores de utilidade. A descrição de como o agente toma decisões e efetua o cálculo da utilidade é apresentada em detalhe na secção 4.2.3.

Atributos	Estrutura de dados
Utilidade mínima	Double
Peso do preço	Double
Peso da quantidade	Double
Peso da data de entrega	Double
Necessidades	Array<Produto> (tabela 4.4)

Tabela 4.3: Estrutura de dados do Modelo do Agente Comprador

O modelo do agente é composto pelas seguintes estruturas de dados: um *double* que representa a utilidade mínima para escolher uma proposta; três *doubles* que representam o peso que o agente comprador atribui a cada um dos atributos em negociação, respetivamente, o atributo preço, o atributo quantidade e o atributo data de entrega. Estes pesos são usados no cálculo da função utilidade (ver 4.2.3); por fim tem um *array* de produtos que o agente pretende adquirir.

O produto (tabela 4.4) é o objeto de negociação, é descrito por um nome, e, para cada atributo de negociação (preço, quantidade e data de entrega), os valores mínimos, máximos e preferidos do agente comprador.

Atributos	Estrutura de dados
nome	String
minPreço	Double
maxPreço	Double
prefPreço	Double
minQuantidade	Double
maxQuantidade	Double
prefQuantidade	Double
minEntrega	Double
maxEntrega	Double
prefEntrega	Double

Tabela 4.4: Classe Produto

4.2.1 Ciclo de vida do agente comprador

Após o arranque do sistema, o agente é inicializado com a informação referido anteriormente: produtos que pretende comprar e os seus respetivos valores máximos, mínimos (que definem o domínio de negociação) e valores preferidos; valores de utilidade mínima; pesos dos diferentes atributos na função de utilidade do agente. Após a inicialização, o agente regista-se no serviço DF da plataforma JADE, procurando e guardando todos os potenciais agentes vendedores que possuem o produto que pretende.

De seguida inicia o processo de negociação (ver figura 4.1), enviando uma mensagem do tipo *Call-For-Proposal* (CFP) para todos os potenciais vendedores. Esta mensagem inclui o nome do produto desejado e os valores mínimos e máximos (domínio de negociação) de cada um dos atributos preço, quantidade e prazo de entrega.

Um exemplo de uma mensagem *Call-For-Proposal* é apresentada a seguir:

CFP: COTTON 1 40 300000 900000 1 20,

Esta mensagem inclui o nome do produto (COTTON), o valor mínimo de preço (1), máximo de preço (40), mínimo de quantidade (300000), máximo de quantidade (900000), mínimo de prazo de entrega (1), máximo de prazo de entrega (20). Estes intervalos de valores têm como objetivo dar um alerta inicial aos agentes vendedores para que estes não façam propostas fora deste intervalo pois serão imediatamente rejeitadas.

De seguida o agente espera pelas respostas dos vendedores que podem ser propostas de venda do produto (*PROPOSE*) ou recusas (*REFUSE*). Assim que forem recebidas todas as respostas à mensagem CFP enviada, são calculadas as utilidades das diferentes propostas usando a função de utilidade do agente (ver 4.2.3). Existe um sistema de *feedback* (ver 4.2.2) onde o agente comprador faz comentários às propostas feitas, no intuito de fazer com que os agentes vendedores melhorem as suas propostas na ronda seguinte. Após envio do *feedback* a cada agente vendedor, o comprador fica à espera de novas propostas ou recusas, repetindo o processo, caso surjam novas propostas. A negociação continua enquanto houver agentes vendedores ativos e enquanto a utilidade média da ronda atual melhora em relação à ronda anterior. É escolhida a proposta com maior utilidade (ver 4.2.3) que ultrapasse a utilidade mínima definida pelo agente comprador. No algoritmo 1 está descrito o ciclo de vida do agente comprador em pseudo-código.

4.2.2 Sistema de *feedback*

Para cada proposta recebida de um agente vendedor, o agente comprador responde com uma mensagem contendo um *feedback* a essa proposta. O *feedback* representa uma avaliação qualitativa à proposta recebida e tem o intuito de auxiliar o vendedor a melhorar a sua proposta na ronda seguinte. A mensagem é do tipo *INFORM* e contém uma avaliação qualitativa para cada um dos atributos em negociação, indicando se o valor desse atributo na proposta está adequado, muito elevado, elevado, baixo ou muito baixo. O *feedback* é dado de acordo com o intervalo de valores

```

Data: Necessidades e utilidades do agente
Result: Melhor proposta
inicialização();
melhorProposta := null;
propostas := null;
utilidadeMedia := 0;
utilidadeMediaUltimaRonda := 0;
vendedores := searchDFSservice(produto);
enviaCFP();
while vendedores.length > 0 do
  propostas := recebeMensagens();
  utilidadeMedia := calculateUtilities(propostas);
  melhorProposta := selecionaMelhorProposta(propostas);
  if utilidadeMedia > utilidadeMediaUltimaRonda then
    enviaFeedBack(propostas);
    utilidadeMediaUltimaRonda := utilidadeMedia;
  else
    if melhorProposta.utilidade >= utilidadeMinima then
      enviaAcceptVencedor();
      enviaRejectPerdedores();
      break;
    else
      // nenhuma proposta alcançou a utilidade mínima;
      break;
    end
  end
end

```

Algorithm 1: Ciclo de vida do comprador

em relação ao valor preferido do agente comprador. Na tabela 4.5 estão identificados os diferentes intervalos e o respetivo *feedback*.

4.2.3 Cálculo da utilidade

A utilidade de uma proposta (equação 4.1) para o agente comprador (μ), é um valor compreendido entre 0 e 1, que representa o valor da proposta para esse agente. O uso do valor utilidade permite também classificar e comparar as diferentes propostas recebidas. Quanto maior for a utilidade de uma proposta mais favorável esta é para o agente.

A utilidade de uma proposta para o agente comprador é definida como a soma de três utilidades parciais 4.1.

$$\mu = \mu_p \times peso_p + \mu_q \times peso_q + \mu_d \times peso_d, \forall \mu : \mu \in [0, 1] \quad (4.1)$$

Onde:

- μ representa a utilidade da proposta.

<i>Feedback</i> dado	Intervalo de valores
MUCHLOWER	valor $\in]pref + 50\% \times (max - pref), max]$
LOWER	valor $\in]pref + 5\% \times (max - pref), pref + 50\% \times (max - pref)[$
OK	valor $\in [pref - 5\% \times (pref - min), pref + 5\% \times (max - pref)]$
HIGHER	valor $\in [pref - 50\% \times (pref - min), pref - 5\% \times (pref - min)[$
MUCHHIGHER	valor $\in [min, pref - 50\% \times (pref - min)[$

Tabela 4.5: *Feedback* dado a cada atributo

- μ_p representa a utilidade do atributo preço.
- $peso_p$ representa o peso do atributo preço.
- μ_q representa a utilidade do atributo quantidade.
- $peso_q$ representa o peso do atributo quantidade.
- μ_d representa a utilidade do atributo data de entrega.
- $peso_d$ representa o peso do atributo data de entrega.

Cada uma das diferentes utilidades parciais, μ_p, μ_q, μ_d , é calculada usando a equação 4.2.

$$\mu_x = 1 - \frac{|valorPref_x - valor_x|}{valorMax_x - valorMin_x} \quad (4.2)$$

Onde:

- μ_x representa a utilidade do atributo x do agente.
- $valorPref_x$ representa o valor preferencial para o atributo x para o agente.
- $valor_x$ representa o valor recebido na proposta para o atributo x .
- $valorMax_x$ representa o valor máximo para o atributo x .
- $valorMin_x$ representa o valor mínimo para o atributo x .

4.3 Agente Vendedor

O agente vendedor é um agente que possui produtos que pretende vender. É responsável por registrar os seus produtos no mercado e aguardar os pedidos do agente comprador. A implementação do agente vendedor, tal como a do agente comprador, inclui o desenvolvimento de duas classes principais relativas ao seu comportamento, *SellerBehaviour* (tabela 4.6) e *SellerDecisionModel* (tabela 4.7).

A classe *SellerBehaviour* implementa o comportamento do vendedor face ao mercado, gere a receção e envio de mensagens. A classe *SellerDecisionModel* é a responsável pelo cálculo dos valores de utilidade e decisão dos níveis de cedência aquando da formulação de novas propostas.

Atributos	Estrutura de dados
Modelo	<i>SellerDecisionModel</i>
CFP	ACLMessage

Tabela 4.6: Estrutura de dados na classe SellerBehaviour

As estruturas de dados mais importantes da classe *SellerBehaviour*, identificadas na tabela 4.6 são: o modelo do agente, que é descrito mais em detalhe de seguida; e a mensagem ACL do tipo *Call-for-proposal*, recebida do comprador, que dá início à negociação.

Atributos	Estrutura de dados
Utilidade mínima	Double
Produtos	Array<Produtos> (tabela 4.4)
Peso do preço	Double
Peso da quantidade	Double
Peso da data de entrega	Double

Tabela 4.7: Estrutura de dados do Modelo do Agente vendedor

O modelo do agente vendedor é similar ao do agente comprador, inclui quatro valores do tipo *doubles* que representam respetivamente, a utilidade mínima para permanecer na negociação, o peso do atributo preço no cálculo da utilidade, o peso do atributo quantidade e o peso do atributo prazo de entrega. Inclui ainda um array de produtos que deseja vender.

4.3.1 Ciclo de vida do agente vendedor

No arranque do sistema o agente vendedor é inicializado com os seus valores de utilidade e produtos que vende. Regista no DF os produtos que comercializa e espera pelo envio da mensagem CFP do agente comprador (pedido de compra). Como referido anteriormente, a mensagem CFP contém os limites mínimos e máximos para os três atributos, desta forma o agente verifica se é possível criar propostas que respeitem a sua utilidade (ver 4.3.2) e os seus limites assim como os limites impostos pelo agente comprador para o domínio da negociação.

Caso os limites impostos não permitam realizar uma proposta com utilidade favorável para o agente, é enviada uma mensagem do tipo REFUSE para o agente comprador e termina a negociação. Caso seja possível respeitar os limites impostos, é calculada uma proposta inicial que é enviada numa mensagem do tipo PROPOSE.

Posteriormente inicia-se o ciclo de *feedback* e consequente cedência, onde o agente espera pelo *feedback* do agente comprador e faz cedências de acordo com as regras estabelecidas na secção 4.3.3. Este ciclo repete-se até o agente receber uma mensagem do tipo ACCEPT_PROPOSAL ou do tipo REJECT_PROPOSAL. No algoritmo 2 encontra-se o ciclo de vida do agente vendedor em pseudo-código.

```

Data: Utilidades e produtos
inicialização();
registerDFService(produtos);
cfp := recebeMensagem();
propostaInicial := criaProposta(cfp);
if propostaInicial then
    enviaProposta();
    while true do
        resposta := recebeMensagem();
        if resposta.performative == REJECT || resposta.performative ==
            ACCEPT_PROPOSAL then
            | break;
        end
        if resposta.performative == INFORM then
            | novaProposta := interpretaFeedback(resposta.getContent());
            | enviaProposta(novaProposta);
        end
    end
else
    | enviaRefuse();
end

```

Algorithm 2: Ciclo de vida do agente vendedor

4.3.2 Utilidade e formação da proposta inicial

A utilidade do agente vendedor é calculada de forma similar à utilidade do agente comprador (ver 4.2.3), baseando-se também em utilidades parciais.

Quando o agente vendedor recebe a mensagem CFP inicial, vai interpretar os três limites impostos pelo comprador: o limite mínimo e máximo do atributo preço, do atributo quantidade e do atributo data de entrega. Para cada limite de cada atributo o agente vai comparar com os seus próprios limites, podendo acontecer sete situações diferentes :

1. Os limites são iguais.
2. Os limites do agente comprador estão contidos nos limites do agente vendedor e não são iguais.
3. Os limites do agente vendedor estão contidos nos limites do agente comprador e não são iguais.
4. O limite inferior do agente comprador é inferior ao limite inferior do vendedor, mas o limite superior está contido nos limites do vendedor.
5. O limite superior do agente comprador é superior ao limite superior do vendedor, mas o limite inferior está contido nos limites do vendedor.
6. O limite inferior do comprador é superior ao limite superior do vendedor.

7. O limite superior do comprador é inferior ao limite inferior do vendedor.

Esta comparação serve para o agente vendedor definir novos intervalos que respeitem quer os seus antigos limites quer os do comprador. Desta forma para cada uma das possíveis situações são calculados novos limites:

1. Os limites não são alterados.
2. Os limites passam a ser os do comprador.
3. Os limites não são alterados.
4. O limite inferior não se altera, o limite superior passa a ser o do comprador.
5. O limite inferior passa a ser o do comprador, o limite superior não se altera.
6. Não é possível haver negociação.
7. Não é possível haver negociação.

Em qualquer um dos casos onde haja alteração dos limites dos intervalos de negociação, se acontecer que o valor preferencial do atributo do agente fique fora do novo intervalo, este valor é alterado para ser igual ao limite inferior, caso esteja mais próximo ao limite inferior, ou ao limite superior, caso contrário. Nos casos onde não é possível haver negociação, é enviado uma mensagem do tipo REFUSE e é terminada a comunicação. Nos casos onde é possível haver negociação, é calculada a utilidade com os valores preferenciais de cada atributo e caso seja superior à utilidade mínima é enviada uma proposta inicial com estes valores, caso seja inferior, é enviado uma mensagem do tipo REFUSE, e cessa a comunicação.

4.3.3 Lidar com *feedback* e cedências

Uma parte importante da negociação é saber como lidar com o *feedback* recebido e decidir as cedências a efetuar. Para tal, foram criadas regras para calcular novos valores de atributos após análise do *feedback* que podem ser encontradas na tabela 4.8.

<i>Feedback</i> recebido	Novo valor para o atributo x
MUCHLOWER	$valorAtual_x - ((valorAtual_x - valorMin_x) \times 20\%)$
LOWER	$valorAtual_x - ((valorAtual_x - valorMin_x) \times 10\%)$
OK	Valor não é alterado
HIGHER	$((valorMax_x - valorAtual_x) \times 10\%) + valorAtual_x$
MUCHHIGHER	$((valorMax_x - valorAtual_x) \times 20\%) + valorAtual_x$

Tabela 4.8: Cedência após *Feedback*

As regras criadas baseiam-se em ceder uma determinada percentagem do intervalo entre o valor atual e um limite. Se o *feedback* sugira um aumento do valor do atributo (HIGHER ou

MUCHHIGHER), esse limite é o superior, caso contrário (LOWER ou MUCHLOWER) esse limite é o inferior.

O agente vai ceder apenas num atributo de cada vez. Para cada atributo, preço, quantidade, prazo de entrega, calcula um novo valor aplicando as fórmulas indicadas na tabela 4.8. De seguida, cria três novas propostas:

- uma proposta onde o novo preço é utilizado e os valores restantes mantêm-se
- uma proposta onde é utilizado o novo valor da quantidade, mantendo-se os restantes atributos
- uma proposta onde é utilizada a nova data de entrega, ficando os outros atributos iguais aos valores antigos.

É então efetuado o cálculo da utilidade de cada uma destas novas proposta e escolhida a que possui maior valor de utilidade. Se o valor de utilidade é igual ou superior ao valor de utilidade mínima do agente vendedor, este envia essa nova proposta para o agente comprador (através da mensagem PROPOSE).

4.4 Resumo

Este capítulo apresenta os detalhes do sistema de negociação sem argumentação, especificando o comportamento dos agentes comprador e vendedor, a forma de comunicação e os tipos de mensagem trocadas.

É também descrito a forma como os agentes tomam decisões, as funções de utilidade, e os ciclos de vida de cada tipo de agente.

No próximo capítulo será apresentado o sistema de negociação com argumentação.

Capítulo 5

Implementação: Sistema com argumentação

5.1 Argumentação

É através da argumentação que os agentes inserem conteúdo adicional à proposta em discussão no mercado, para tentar alcançar um acordo. Um argumento é uma estrutura composta por duas partes, uma pretensão e uma razão. Neste caso, e do ponto de vista do agente comprador, a pretensão representa a proposta a aceitar por este agente e a razão a justificação para a aceitar. Por exemplo, um argumento poderá ser "preço tem de ser inferior a 20, pois existe uma proposta melhor", sendo que "preço tem de ser inferior a 20" é a pretensão e "existe uma proposta melhor" é a razão.

Neste sentido, foi definido que, no mercado considerado neste trabalho, o agente comprador poderá utilizar dois tipos de argumentos: apelo ao interesse próprio e apelo a que promessas passadas sejam cumpridas. O agente vendedor poderá utilizar também dois tipos de argumentos: promessas e recompensas. De seguida, explica-se melhor o significado de cada um destes argumentos, que se encontram ordenados por ordem crescente de agressividade.

- Apelo a que promessas passadas sejam cumpridas - Este argumento, que pode ser utilizado pelo agente comprador, tem como objetivo lembrar ao agente com quem se está a negociar que este fez uma promessa no passado e a deverá cumprir, cessando a negociação caso não a cumpra.
- Apelo ao interesse próprio - Este argumento, utilizado apenas pelo agente comprador, é usado para lembrar ao agente vendedor que é do seu interesse aceitar uma determinada proposta apresentando a razão para tal.
- Recompensas - Este argumento, utilizado apenas pelos agentes vendedores, tem como objetivo atribuir uma recompensa imediata ao agente comprador, caso este aceite a sua proposta.

Implementação: Sistema com argumentação

- Promessas - Este argumento, que pode ser utilizado por agentes vendedores tem como objetivo fazer promessas(a cumprir no futuro) ao agente com qual se está a negociar.

No contexto desta dissertação foram definidas as seguintes pretensões e razões:

Pretensões:

- Valor do preço
- Valor da data de entrega
- Valor da quantidade

A pretensão representa o atributo da proposta que tem de ser melhorado ou está a ser alvo de negociação.

Razões:

- PROMISE_MADE - Utilizado pelo agente comprador, representa o apelo a que as promessas passadas sejam cumpridas. É enviado a agentes vendedores que tenham feito promessas no passado.
- BETTER_PROPOSAL - Utilizado para representar o argumento de apelo ao interesse próprio. É utilizado pelo agente comprador e enviado aos agentes vendedores que não têm a melhor proposta, para os avisar que estes não têm a melhor proposta e por isso a têm de melhorar ou não ganharam a negociação.
- REWARD - Utilizado pelos agentes vendedores, representa recompensas. É utilizado quando não pretendem alterar o valor proposto pelo agente comprador, mas estão disponíveis a ceder noutros atributos como recompensa.
- PROMISE - Utilizado pelos agentes vendedores, representa promessas. É usado quando não pretende alterar o valor proposto pelo comprador e este também não aceita recompensas. A promessa não é sobre o produto em negociação, mas noutro que o vendedor vende e o comprador quer comprar.

Exemplos de mensagens ilustrativas do uso de cada um destes argumentos, são apresentados a seguir:

- Apelo a que promessas passadas sejam cumpridas

PRICE LOWER 10 PROMISE_MADE PRICE LOWER 10

significado: "preço tem de ser inferior a 10 pois numa promessa passada prometeste-me um preço inferior a 10".

- Apelo ao interesse próprio

QUANTITY HIGHER 300000 BETTER_PROPOSAL significado: "a quantidade de produto tem de ser superior a 300000 pois existe uma proposta melhor que a tua que tem esse valor"

- Recompensas

PRICE 4 REWARD DELIVERY 5

significado: "caso aceites que o preço se mantenha em 4, cedo na data de entrega para 5"

- Promessas

PRICE 7 PROMISE COTTON QUANTITY 900000

significado: "caso aceites que o preço se mantenha em 7, no futuro prometo-te uma quantidade de 900000 num carregamento de algodão"

Todo o sistema de negociação com argumentação foi implementado por cima do sistema sem argumentação (descrito no capítulo 4), partilhando por isso muitas das soluções já descritas.

5.2 Comunicação entre agentes

Para além de todos os tipos de mensagem mencionados no sistema de negociação sem argumentação (ver 4.1) foi ainda utilizado o tipo **INFORM_IF**:

- **INFORM_IF** - Este tipo de mensagem é utilizado tanto pelo agente comprador como pelo agente vendedor, para indicar que o conteúdo da mensagem é um argumento.

A figura 5.1 descreve o fluxo de comunicação do sistema com argumentação.

5.3 Agente Comprador

O agente comprador é idêntico ao agente comprador do sistema base sem argumentação, partilhando por isso das mesmas funções de utilidade e estruturas de dados. A exceção é a adição de uma estrutura que guarda informação de todas as promessas feitas no passado ao agente.

5.3.1 Ciclo de vida do agente comprador

Tal como no sistema sem argumentação, após o arranque do sistema, o agente é inicializado com a informação sobre os produtos que pretende comprar, os seus respetivos valores máximos, mínimos e preferidos, bem como os valores de utilidade mínima e os pesos dos diferentes atributos na função de utilidade do agente. De seguida, o agente comprador regista-se no serviço DF, procurando e guardando todos os potenciais agentes vendedores que têm o produto que procura.

Formulação do pedido de compra

Para cada um dos potenciais agentes vendedores, o agente envia uma mensagem do tipo *Call-For-Proposal* contendo o nome do produto e os domínios de negociação e aguarda pelas propostas iniciais dos agentes vendedores, iniciando-se assim a negociação.

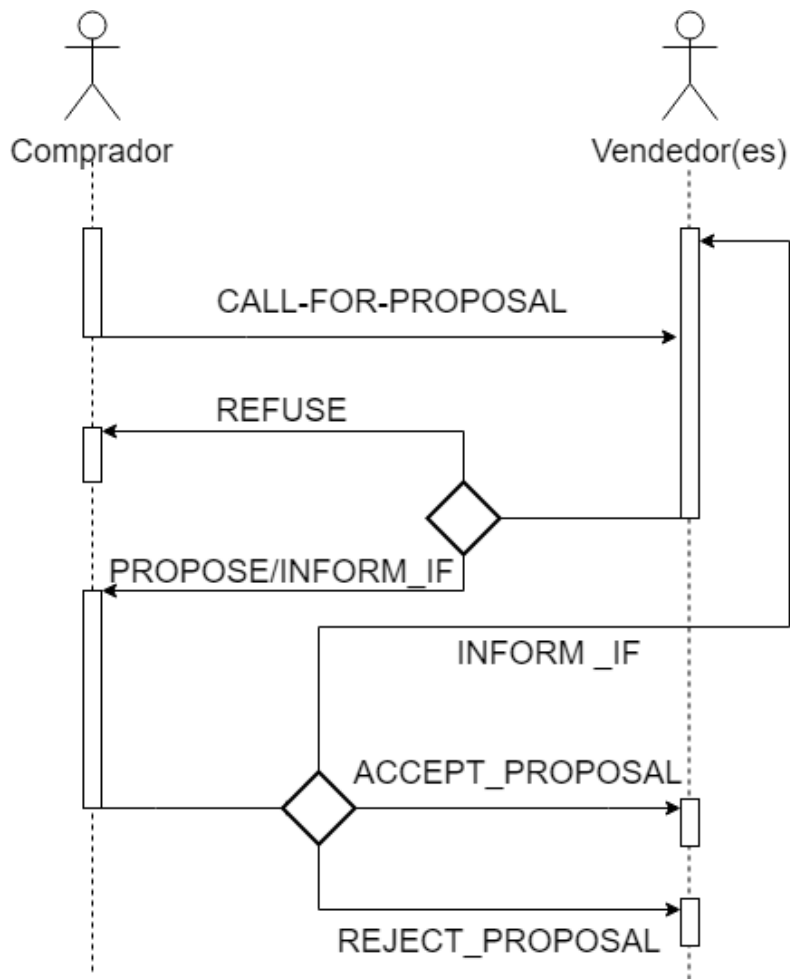


Figura 5.1: Diagrama de comunicação - Sistema com argumentação

Os agentes utilizam uma estratégia de argumentação que consiste em utilizar primeiro os argumentos com menor agressividade. No caso do agente comprador, o apelo ao cumprimento de promessas passadas possui menor agressividade que o apelo ao interesse próprio.

Receção de propostas dos agentes vendedores

Após receber as propostas iniciais, o agente comprador vai verificar se existe alguma promessa feita anteriormente por algum agente ativo na negociação.

Caso exista uma promessa passada, e seja adequada para a negociação atual, é enviado um apelo a que essa promessa seja cumprida (algoritmo 3), sendo enviada uma mensagem ACL com *performative* **INFORM_IF**, cujo conteúdo será o valor do atributo (pretensão) que o agente prometeu e a razão **PROMISE_MADE**. Todos os outros agentes vendedores que não tenham feito promessas são mantidos em espera. Após enviada esta mensagem, o agente comprador aguarda que o agente vendedor responda com nova proposta que cumpra a promessa feita. Se a nova proposta não for satisfatória, o agente comprador rejeita e cessa a negociação com esse vendedor,

rejeitando todas as possíveis promessas no futuro.

Caso não existam promessas passadas, o agente comprador passa a utilizar o argumento de apelo ao interesse próprio. Para tal, o agente verifica de entre todo o conjunto de propostas recebidas qual é a que tem maior valor de utilidade. Posteriormente, para todas as outras propostas, verifica qual é o atributo dessa proposta que tem de ser melhorado em relação à melhor proposta. Esta escolha é realizada por ordem de peso dos atributos, ou seja, começa pelo atributo que tem mais peso na utilidade e acaba no que tem menor peso. Por exemplo, caso $pesoPreço > pesoQuantidade > pesoEntrega$, o agente vai começar por verificar se o preço da proposta é maior do que o preço da melhor proposta, se for, envia uma mensagem com o *performative* **INFORM_IF**, cujo conteúdo é o valor do preço da melhor proposta e a razão é **BETTER_PROPOSAL**, caso o preço não necessite de ser alterado (por ser igual ou até melhor do que o da melhor proposta), o agente verifica a quantidade e posteriormente a data de entrega.

Após envio das mensagens com os argumentos, o agente espera por novas propostas dos vendedores, analisando-as e tentando utilizar os argumentos disponíveis para tentar que os vendedores cedam mais. Este ciclo repete-se até não ser possível utilizar nenhum argumento, não haver mais alterações aos valores dos atributos ou só restar um vendedor ativo na negociação.

De referir que o agente comprador pode receber recompensas e promessas por parte dos agentes vendedores. A forma de lidar com promessas e recompensas está descrita na secção 5.3.2.

5.3.2 Lidar com recompensas e promessas

O agente comprador pode receber como argumentos, recompensas e promessas dos agentes vendedores.

Recompensa

As recompensas são uma resposta a um apelo ao interesse próprio enviado anteriormente pelo agente comprador. Ao receber uma recompensa, o agente comprador vai calcular a sua utilidade. Caso a utilidade da recompensa seja superior à da melhor proposta até ao momento, ele aceita-a. Caso a utilidade seja inferior, ele rejeita-a, enviando uma mensagem do tipo **REJECT_PROPOSAL**.

Promessa

Apesar de numa negociação só ser negociado um produto, o agente comprador pode ter intenção de adquirir outros produtos em futuras negociações. Desta forma, o agente pode receber promessas relativas a esses outros produtos. Ao contrário de uma recompensa, ao aceitar uma promessa, a negociação termina.

Ao receber uma promessa, o agente vai verificar se no seu *array* de necessidades se encontra o produto referido na promessa. Caso não exista, envia uma mensagem do tipo **REJECT_PROPOSAL** e rejeita a promessa. Caso exista, aceita a promessa apenas se forem cumpridos os três requisitos seguintes:

- O atributo prometido é o que tem maior peso na sua função de utilidade
- O valor do atributo prometido está perto do valor preferido (um intervalo de 5%).
- A utilidade da proposta atual não é muito inferior à utilidade da melhor proposta (pode ser até 10% inferior).

Caso um destes requisitos não se cumpra, a promessa é rejeitada. Se for aceite é enviada uma mensagem do tipo **ACCEPT_PROPOSAL** para o agente que fez a promessa e é terminada a negociação.

Data: Propostas e Promessas Passadas

argumentosEnviados := 0;

mensagensRecebidas := 0;

foreach *proposta* in *Propostas* **do**

promessa := *promessasPassadas*.existe(*sellerID*, *proposta*.nomeProduto);

if *promessa* != *null* **then**

cumprePromessa := *avaliaPromessa*(*promessa*, *proposta*);

if !*cumprePromessa* **then**

 enviaArgumento(*sellerID*, PROMISE_MADE, *promessa*);

 argumentosEnviados ++ ;

end

end

end

//espera pela resposta

while *mensagensRecebidas* != *argumentosEnviados* **do**

proposta = recebeMensagem();

if *proposta* != *null* **then**

mensagensRecebidas ++ ;

promessa := *promessasPassadas*.get(*sellerID*, *proposta*.nomeProduto);

cumprePromessa := *avaliaPromessa*(*promessa*, *proposta*);

if !*cumprePromessa* **then**

 enviaReject(*sellerID*);

vendedoresAtivos.remove(*sellerID*);

end

end

end

Algorithm 3: Algoritmo de verificação de promessas - Agente Comprador

5.4 Agente Vendedor

O agente vendedor é idêntico ao agente vendedor no sistema base sem argumentação, possui as mesmas funções de utilidade e estruturas de dados. A exceção é a adição de uma estrutura para guardar informação de todas as promessas realizadas pelo agente.

5.4.1 Ciclo de vida do agente vendedor

Tal como no sistema sem argumentação, o agente vendedor é passivo. Após ser inicializado com os seus valores de utilidade e os produtos que pretende vender, regista-se no DF e aguarda pela mensagem *Call-For-Proposal* do comprador (pedido de compra). Após análise da mensagem CFP, o agente vendedor envia uma proposta inicial utilizando a mesma estratégia descrita na secção 4.3.2 e aguarda pela resposta do agente comprador.

Neste sistema com argumentação a resposta poderá ser de dois tipos, um apelo a que promessas sejam cumpridas ou um apelo ao interesse próprio.

A forma de lidar com os apelos recebidos e fazer novas propostas encontra-se descrita na secção 5.4.2.

Se o agente vendedor consegue encontrar uma nova proposta, esta é enviada ao agente comprador e aguarda pela resposta. Este processo é iterativo e repete-se até não ser possível ao vendedor ceder mais, ou até receber uma mensagem informando que a sua proposta foi aceite.

5.4.2 Lidar com apelos

O agente vendedor é um agente que irá tentar satisfazer os pedidos do agente comprador, desde que consiga respeitar as suas utilidades mínimas.

Apelo ao cumprimento de promessas passadas

Quando o vendedor recebe um apelo ao cumprimento de promessas passadas, o agente vai verificar se a razão do apelo é verdadeira, pesquisando no seu histórico de promessas feitas se existe alguma promessa realizada àquele comprador, verificando de seguida se a pretensão é verdadeira também, comparando a informação do seu histórico com a informação recebida, caso se verifique que o apelo é verdadeiro, o vendedor faz as cedências necessárias para cumprir a promessa feita. Caso seja falsa, o agente vendedor envia uma mensagem do tipo **REFUSE**, terminando a negociação.

Apelo ao interesse próprio

Quando o vendedor recebe um apelo ao interesse próprio, vai acreditar na informação enunciada e ceder para tentar que a sua proposta seja aceite. Verifica se o valor do atributo presente no apelo está dentro do seu domínio de negociação. Caso esteja, vai ceder segundo a tabela 5.1 caso o atributo seja o que tem mais peso na sua função de utilidade, ou segundo a tabela 5.2 caso contrário.

Resposta recebida	Novo valor para o atributo x
LOWER	$valorAtual_x - ((valorAtual_x - valorMin_x) \times 10\%)$
HIGHER	$((valorMax_x - valorAtual_x) \times 10\%) + valorAtual_x$

Tabela 5.1: Cedência caso o atributo seja importante

Resposta recebida	Novo valor para o atributo x
LOWER	$valorAtual_x - ((valorAtual_x - valorMin_x) \times 20\%)$
HIGHER	$((valorMax_x - valorAtual_x) \times 20\%) + valorAtual_x$

Tabela 5.2: Cedência caso o atributo seja não importante

Caso não seja possível ceder mais no atributo referido no apelo, mas o agente comprador continuar a informar que existe uma proposta melhor, o agente tentará enviar uma recompensa, cedendo noutro atributo. As recompensas neste contexto poderão ser interpretadas da seguinte maneira: "Se aceitar que eu não ceda neste atributo, ofereço mais noutro atributo".

Neste sentido, o vendedor cede no atributo menos importante de entre os dois restantes, tentando prever um valor que será benéfico para o comprador: caso o atributo a ceder seja o preço, o agente envia o menor preço possível dentro das suas utilidade; caso seja a quantidade, envia o máximo de quantidade possível; caso seja o prazo de entrega, envia como recompensa a menor data de entrega possível. Se não for possível ceder no atributo menos importante, o agente tenta ceder no restante atributo.

Se mesmo oferecendo recompensas, o agente comprador rejeitar, o vendedor pode fazer promessas. O vendedor só fará promessas se vender outros produtos e tal como nas recompensas, o vendedor tenta prever um valor que será benéfico para o comprador.

Por exemplo, considere-se um vendedor que vende algodão e lenços, sendo a negociação atual sobre o produto algodão. O vendedor recebe um argumento dizendo que existe uma proposta melhor com quantidade de 800000. No entanto, o vendedor não pode satisfazer essa quantidade em algodão, e o comprador também rejeita as recompensas noutros atributos. Por isso o vendedor tenta oferecer como promessa uma quantidade de 800000 de lenços na próxima negociação com o comprador.

5.5 Resumo

Este capítulo apresenta os detalhes do sistema com argumentação, especificando o comportamento dos agentes compradores e vendedores, a forma de comunicação e os tipos de mensagens trocadas. É descrito o conceito de argumentação no contexto desta dissertação e identificados os tipos de argumentos que os agentes podem utilizar. É também descrito o processo de tomada de decisão e ações dos agentes.

Capítulo 6

Experimentação e resultados

Uma vez implementado o sistema, é necessário que sejam feitas algumas experiências para o validar e testar. Serão estudados quatro cenários e realizadas comparações entre o sistema de negociação sem argumentação e o sistema de negociação com argumentação. Cada cenário será devidamente caracterizado, descrevendo o objeto de negociação, os diferentes agentes na negociação e respectivos domínios de negociação e valores preferidos.

6.1 Cenário 1

Neste cenário, o sistema possui três agentes, um comprador e dois vendedores. A negociação é sobre um carregamento de algodão. Com este cenário pretende-se analisar o comportamento de agentes vendedores com domínios de negociações idênticos e sem promessas por cumprir, quando a negociação ocorre sem e com argumentação. Os domínios de negociação para cada um dos agentes são descritos de seguida.

Agente comprador: procura algodão

- Preço mínimo: 1; Preço máximo: 10; Preço preferido: 1
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 200000
- Data de entrega mínima: 7; Data de entrega máxima: 31; Data de entrega preferida: 10
- Utilidade mínima: 0.5; Peso preço: 0.2, Peso quantidade: 0.3; Peso Entrega: 0.5

Agente vendedor 1: vende algodão

- Preço mínimo: 1; Preço máximo: 10; Preço preferido: 3
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 250000
- Data de entrega mínima: 7; Data de entrega máxima: 31; Data de entrega preferida: 12
- Utilidade mínima: 0.5; Peso preço: 0.15, Peso quantidade: 0.35; Peso Entrega: 0.5

Experimentação e resultados

Agente vendedor 2: vende algodão

- Preço mínimo: 1; Preço máximo: 10; Preço preferido: 3
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 250000
- Data de entrega mínima: 7; Data de entrega máxima: 31; Data de entrega preferida: 12
- Utilidade mínima: 0.5; Peso preço: 0.15, Peso quantidade: 0.35; Peso Entrega: 0.5

Neste cenário não existem promessas passadas por cumprir e os agentes podem utilizar todos os argumentos. Como se pode constatar, o agente vendedor 1 e o agente vendedor 2 têm os mesmos valores de preferência e domínios de negociação.

6.1.1 Negociação sem argumentação

Os gráficos 6.1 e 6.2 apresentam os resultados da negociação no sistema sem argumentação do ponto de vista dos vendedores e comprador, respetivamente.

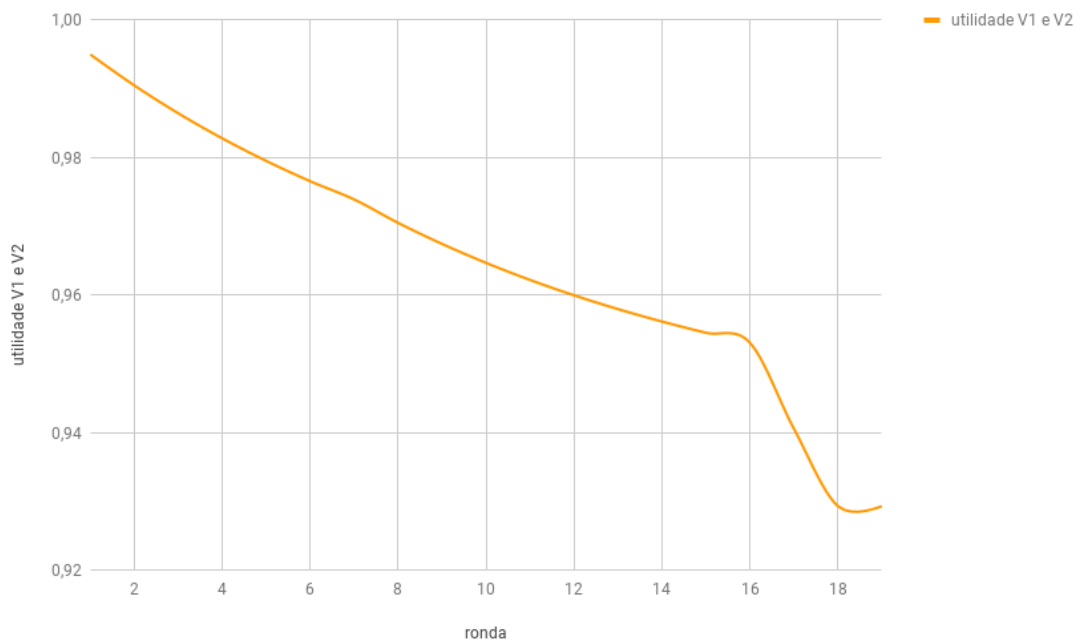


Figura 6.1: Cenário 1: Negociação sem argumentação - Utilidade dos vendedores

Como pode ser observado pelo gráfico 6.1, a utilidade dos vendedores e o comportamento demonstrado são iguais para ambos, pois os seus domínios de negociação e importância dos atributos em negociação (pesos dos atributos) são iguais.

Os agentes vendedores começam por enviar uma proposta que tem o valor de utilidade máximo para eles e vão sucessivamente cedendo em conformidade com o *feedback* do agente vendedor. A mesma análise pode ser feita sobre o gráfico 6.2, que mostra a utilidade da melhor proposta

Experimentação e resultados

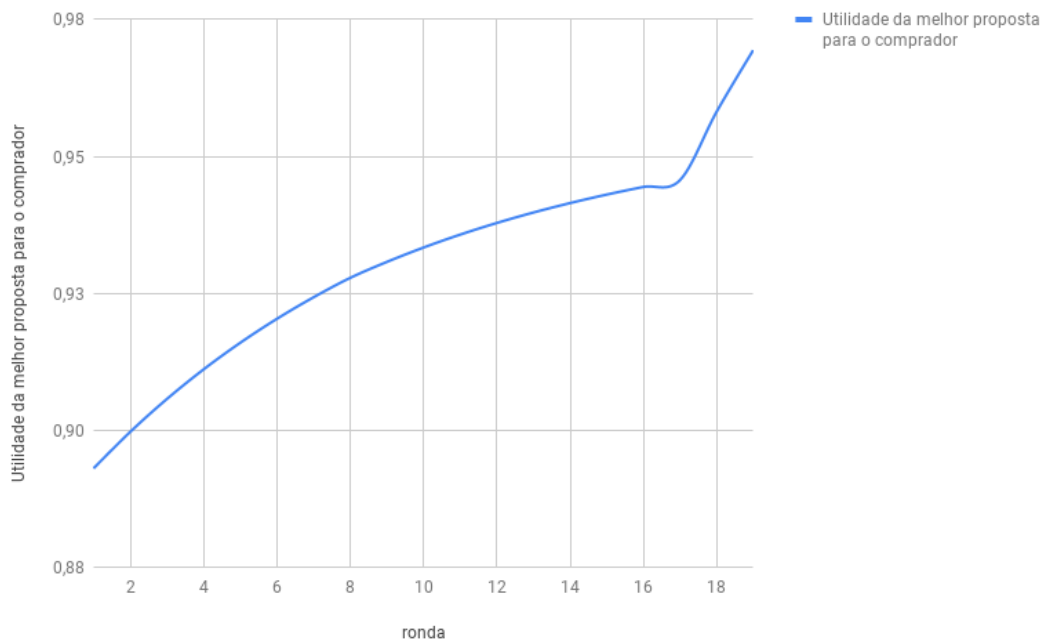


Figura 6.2: Cenário 1: Negociação sem argumentação - Utilidade comprador

recebida pelo agente comprador. Neste gráfico podemos ver que o comprador recebe uma proposta inicial com valor de utilidade de 0.89 e ao longo da negociação este valor vai sendo melhorado até chegar a 0.97 aproximadamente.

6.1.2 Negociação com argumentação

Este cenário trata um caso especial para o sistema de negociação com argumentação, o caso em que as propostas iniciais dos agentes vendedores são iguais e não existem promessas por cumprir.

Nesta situação particular não ocorre nenhum tipo de argumentação. Os agentes vendedores começam por enviar as propostas iniciais, que são iguais, e de seguida o agente comprador tenta negociar com argumentação. O comprador verifica se existem promessas por realizar, que não existem. Tenta então usar o argumento de apelo ao interesse próprio, mas não pode utilizar a justificação de que existe uma proposta melhor, pois as propostas iniciais são iguais. O comprador acaba apenas por escolher uma de entre as duas propostas como sendo a vencedora. Neste caso a utilidade da proposta inicial é bastante elevada, cerca de 0.89, e maior do que a utilidade mínima do agente, sendo assim possível chegar a acordo, sendo que só existe uma ronda e não ocorre argumentação. No entanto, se a proposta inicial tivesse utilidade menor do que a utilidade mínima do agente comprador, não seria possível melhorar esse valor, acabando a negociação por fracassar.

Para este situação particular, a negociação com argumentação pode não ser satisfatória.

6.2 Cenário 2

Nesta negociação o sistema envolve três agentes, um comprador e dois vendedores. A negociação é sobre um carregamento de lenços. Este cenário tem como objetivo analisar o comportamento de agentes vendedores com domínios de negociações diferentes e sem promessas por cumprir, sendo só possível utilizar os apelos pelo agente comprador, enquanto os agentes vendedores não podem utilizar nem promessas nem recompensas. Os domínios de negociação para cada agente são descritos de seguida.

Agente comprador: procura lenços

- Preço mínimo: 1; Preço máximo: 10; Preço preferido: 1
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 900000
- Data de entrega mínima: 1; Data de entrega máxima: 31; Data de entrega preferida: 1
- Utilidade mínima: 0.5; Peso preço: 0.2, Peso quantidade: 0.3; Peso Entrega: 0.5

Agente vendedor 1: vende lenços

- Preço mínimo: 1; Preço máximo: 10; Preço preferido: 5
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 820000
- Data de entrega mínima: 3; Data de entrega máxima: 31; Data de entrega preferida: 7
- Utilidade mínima: 0.5; Peso preço: 0.15, Peso quantidade: 0.5; Peso Entrega: 0.35

Agente vendedor 2: vende lenços

- Preço mínimo: 1; Preço máximo: 7; Preço preferido: 4
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 700000
- Data de entrega mínima: 4; Data de entrega máxima: 31; Data de entrega preferida: 10
- Utilidade mínima: 0.5; Peso preço: 0.5, Peso quantidade: 0.35; Peso Entrega: 0.15

6.2.1 Sistema sem argumentação

O gráfico 6.3 mostra os valores de utilidade das propostas para o agente comprador e o gráfico 6.4 mostra a utilidade de cada vendedor. Esta negociação terminou na ronda trinta.

No gráfico 6.3 vê-se que o comprador recebe duas propostas em que as utilidades iniciais são de aproximadamente 0.78 e 0.70. Seguidamente começa o ciclo de *feedback* que leva a que as propostas sejam melhoradas, embora lentamente.

Por volta da ronda 11, para o vendedor 2, e da ronda 14 para o vendedor 1, existe uma desaceleração e conseqüente aceleração da cedência nas propostas. A desaceleração acontece pois um

Experimentação e resultados

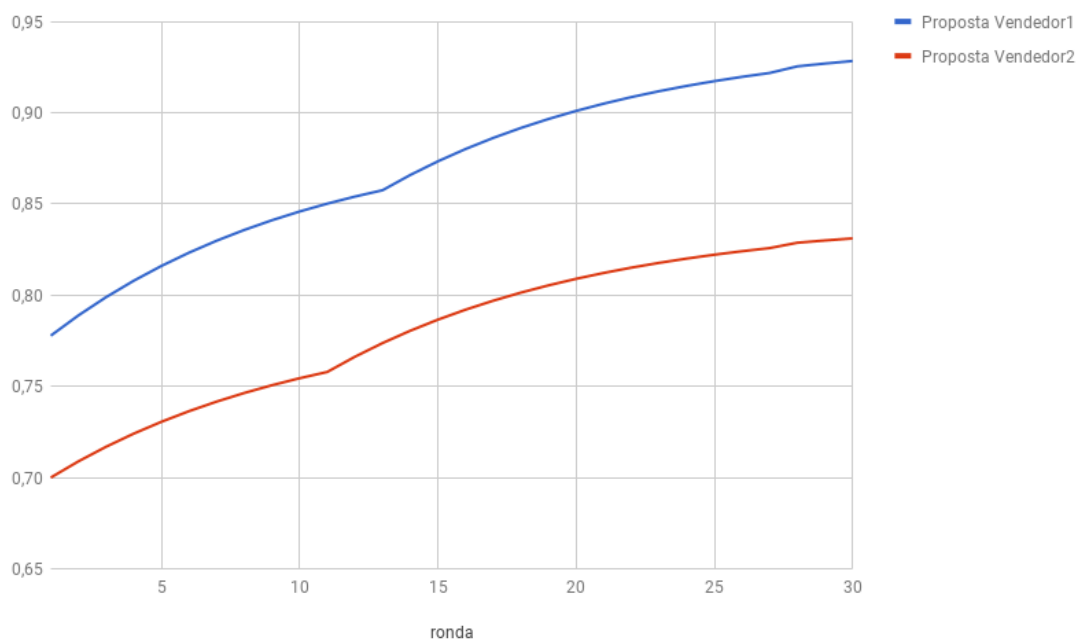


Figura 6.3: Cenário 2: Negociação sem argumentação - Utilidade comprador

atributo da proposta do agente vendedor está prestes a chegar a um valor aceitável pelo agente comprador. A aceleração é uma consequência do anterior, após um valor de atributo ser aceitável, o agente inicia a cedência noutra atributo, e a estratégia do agente vendedor leva a que a cedência seja mais elevada no início e tenda a estabilizar, como se pode verificar no gráfico 6.4.

No geral, a utilidade das propostas para o agente comprador vão aumentando de forma constante. A proposta do vendedor 1 é melhorada de 0.78 para cerca de 0.93, um aumento de 0.15. A proposta do vendedor 2 inicia-se com utilidade de 0.7 e termina com utilidade de 0.83, um aumento de 0.13. No final foi escolhida a proposta do vendedor 1 pois é a que tem melhor utilidade.

6.2.2 Sistema com argumentação

Nas figuras 6.5 e 6.6 encontram-se, respetivamente, os valores de utilidade das propostas para o comprador e os valores de utilidade para os vendedores. Esta negociação chegou ao fim na décima oitava ronda.

A negociação começa com propostas iniciais com valor de utilidade para o agente comprador de 0.78 para a proposta do vendedor 1, e 0.7 para o vendedor 2. Uma primeira diferença relativamente ao sistema de negociação sem argumentação é que as utilidades, tanto do agente comprador (ver figura 6.5) como dos agentes vendedores (ver figura 6.6) não variam de forma suave, mas sim de forma repentina, com picos súbitos.

De forma a compreender melhor a razão destas variações são apresentadas algumas mensagens trocadas e ações realizadas durante as rondas negociais.

Experimentação e resultados

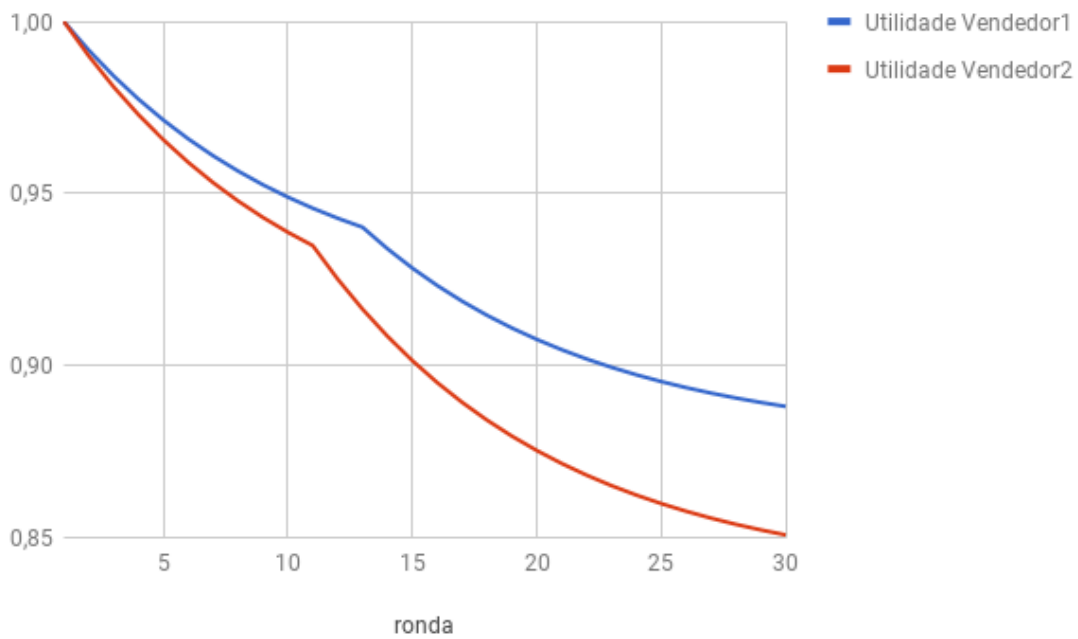


Figura 6.4: Cenário 2: Negociação sem argumentação - Utilidade dos vendedores

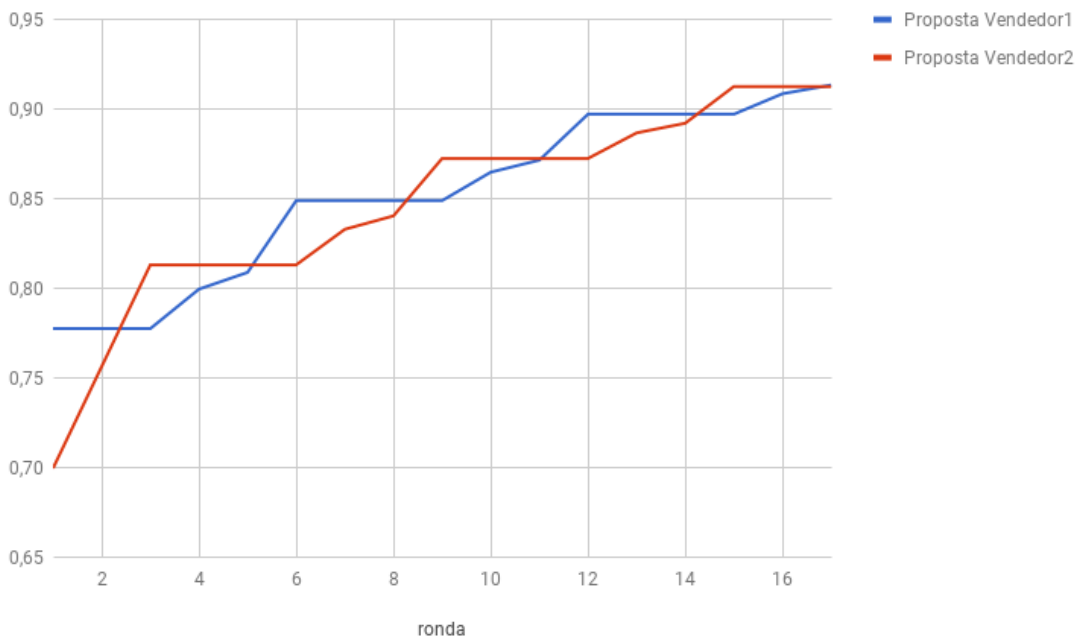


Figura 6.5: Cenário 2: Negociação com argumentação - Utilidade comprador

P representa Preço, Q representa Quantidade e E representa Data de Entrega Ronda 1

Experimentação e resultados

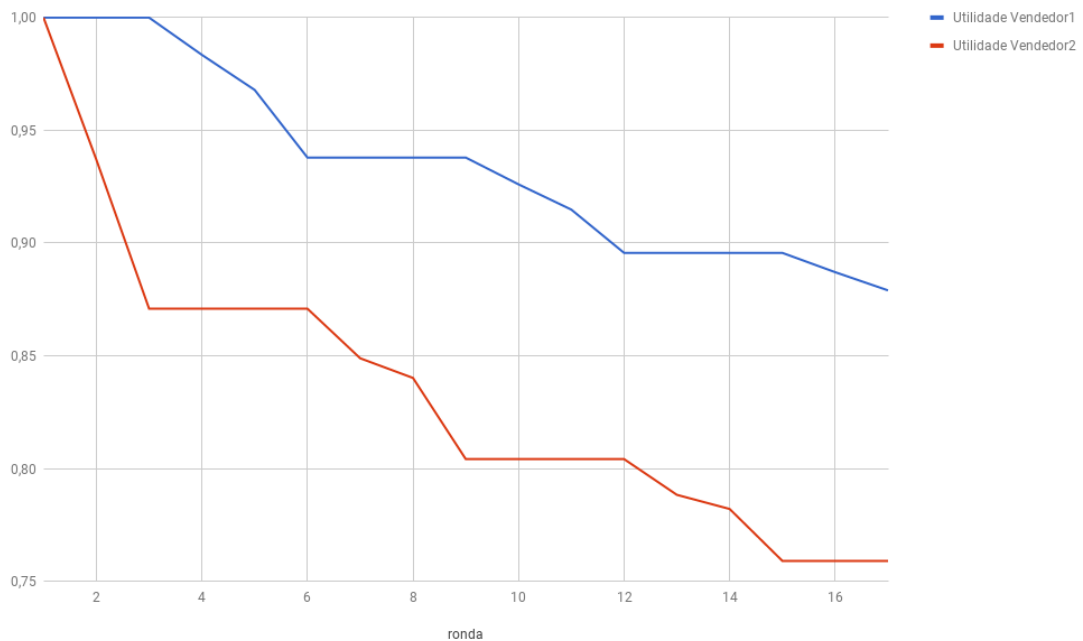


Figura 6.6: Cenário 2: Negociação com argumentação - Utilidade dos vendedores

Comprador envia dois CFP: Voile 1.0 10.0 180000.0 900000.0 1.0 31.0

Comprador recebe: PROPOSE P:5.0 Q:820000.0 E:7.0 do vendedor 1 com utilidade de 0.778

Comprador recebe: PROPOSE P:4.0 Q:700000.0 E:10.0 do vendedor 2 com utilidade de 0.7

Ronda 2

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia: INFORM_IF BETTER_PROPOSAL DELIVERY 7.0 para o vendedor 2. Pois é o vendedor 2 que tem a pior proposta.

Comprador recebe: PROPOSE P:4.0 Q:700000.0 E:6.6 do vendedor 2 com utilidade 0.756

Ronda 3

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia: INFORM_IF BETTER_PROPOSAL QUANTITY 820000.0 para o vendedor 2. Pois o vendedor 2 continua com a pior proposta.

Comprador recebe: PROPOSE P:4.0 Q:836000.2 E:6.6 do vendedor 2 com utilidade 0.813

Experimentação e resultados

Ronda 4

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia: `INFORM_IF BETTER_PROPOSAL DELIVERY` para o vendedor 1.

Comprador recebe: `PROPOSE P:5.0 Q:820000.0 E:5.68` do vendedor 1 com utilidade 0.79977

....

Ronda 18

Propostas atuais: Vendedor 1: `P:2.048 Q:870140 E:4.045312` com utilidade de 0.9135145.

Vendedor 2: `P:1.6384 Q:866822.94 E:4.55664` com utilidade de 0.9127122

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia: `INFORM_IF BETTER_PROPOSAL DELIVERY 4.045312` para o vendedor 2.

Como o valor mínimo de data de entrega (delivery) para o vendedor 2 é 4.0 e visto não poder utilizar recompensas neste cenário, o vendedor não pode ceder mais e envia um `REFUSAL`.

Sendo que o vendedor 1 ganha a negociação.

Uma diferença entre o sistema de negociação com argumentação e sem argumentação é que no sistema com argumentação existem rondas em que os agentes vendedores ficam em espera, não recebendo nem enviando mensagens. Nos gráficos 6.5 e 6.6 é fácil identificar essas rondas, pois a utilidade mantém-se constante. Analisando as mensagens trocadas, após receber as propostas iniciais, o agente comprador envia para o agente vendedor que não tem a melhor proposta, um apelo ao interesse próprio (para que este melhore a sua proposta na ronda seguinte, caso contrário não ganhará a negociação) onde refere o atributo que tem de melhorar e aguarda uma nova proposta. Este ciclo repete-se até o agente vendedor não conseguir ceder mais num qualquer atributo.

Neste sistema, a argumentação de apelo ao interesse próprio permite ao agente comprador fazer com que os vendedores entrem em competição entre eles, fazendo com que melhorem as suas propostas em relação aos outros, ao contrário do sistema sem argumentação, onde a proposta é melhorada, mas sem noção dos adversários. Esta é a razão das variações bruscas nos valores de utilidade. No gráfico 6.5 é claro perceber a competição, por exemplo, nas rondas 3, 4 e 5, o agente vendedor 2 tem a melhor proposta, mas nas rondas seguintes é ultrapassado pelo vendedor 1.

Os valores de utilidade para o comprador são melhorados de 0.78 para 0.9135 (propostas do vendedor 1) e de 0.7 para 0.9127 (propostas do vendedor 2), um aumento de 0.134 e de 0.21 em apenas 18 rondas.

Experimentação e resultados

Neste cenário, a utilidade da proposta vencedora, tanto para o agente comprador como para o vendedor que ganhou a negociação, é bastante idêntica entre os sistemas com e sem argumentação. No sistema sem argumentação a utilidade para o comprador é de 0.93 e para o vendedor é de 0.89, enquanto que no sistema com argumentação as utilidades são de 0.91, para o comprador, e de 0.88, para o vendedor que ganha. A distinção entre os sistemas encontra-se no número de rondas e na diferença dos valores de utilidade das diferentes propostas para o agente comprador.

No caso do sistema sem argumentação a diferença de utilidade para o comprador entre a proposta vencedora e a que ficou em segundo lugar é de cerca de 0.10 enquanto no sistema com argumentação é praticamente nula, como referido anteriormente esta diferença deve-se à competição causada pela argumentação.

6.3 Cenário 3

Este cenário é idêntico ao Cenário 2 mas tem como principal objetivo analisar o comportamento dos agentes quando é permitido o uso de recompensas, sendo por isso permitido o uso deste argumento.

6.3.1 Negociação sem argumentação

Como se trata de um cenário idêntico ao anterior, onde a única diferença se encontra nos argumentos usados, a análise do processo de negociação sem argumentação é igual à do cenário 2 (secção 6.2.1).

6.3.2 Negociação com argumentação

Como o cenário é idêntico ao cenário 2, apenas com a diferença que os agentes vendedores podem utilizar o argumento de recompensa, toda a negociação acontece nos mesmos termos até à ronda 18. Nesta ronda, no cenário anterior, a negociação termina com utilidade para o comprador de 0.9135, mas neste cenário a negociação prolonga-se até à ronda 22 que termina com utilidade para o comprador de 0.95. Para uma melhor análise do processo de negociação, são mostradas de seguida as mensagens trocadas entre os agentes e ações realizadas por estes a partir da ronda 18:

P representa Preço, Q representa Quantidade e E representa Data de Entrega

Ronda 18

Propostas atuais: Vendedor 1: P:2.048 Q:870140 E:4.045312 com utilidade de 0.91351455

Vendedor 2: P:1.6384 Q:866822.94 E:4.55664 com utilidade de 0.9127122

Experimentação e resultados

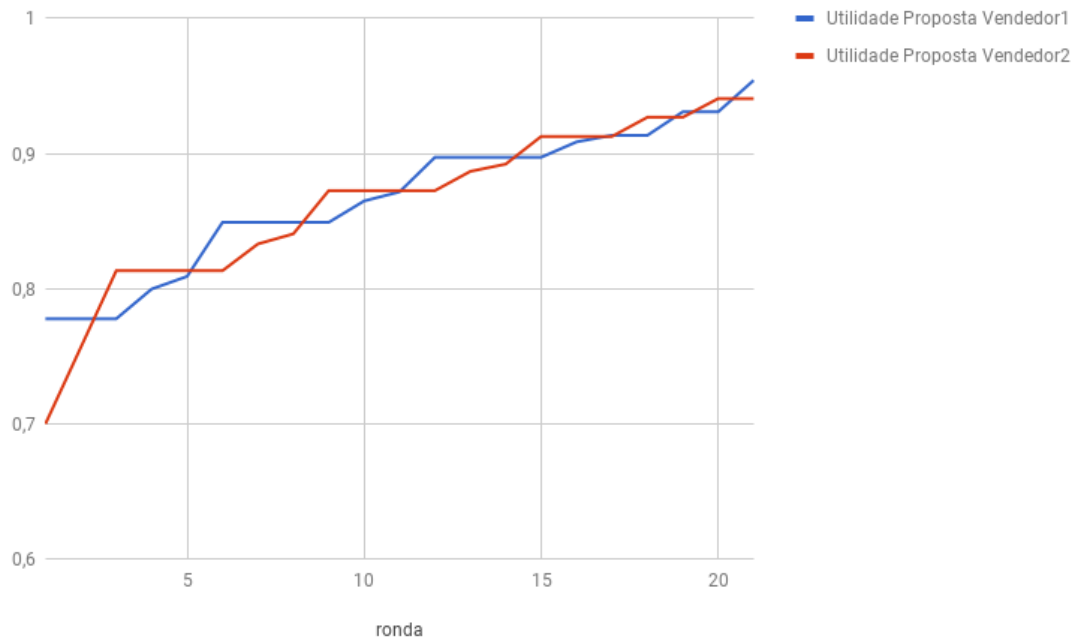


Figura 6.7: Cenário 3: Negociação com argumentação - Utilidade comprador

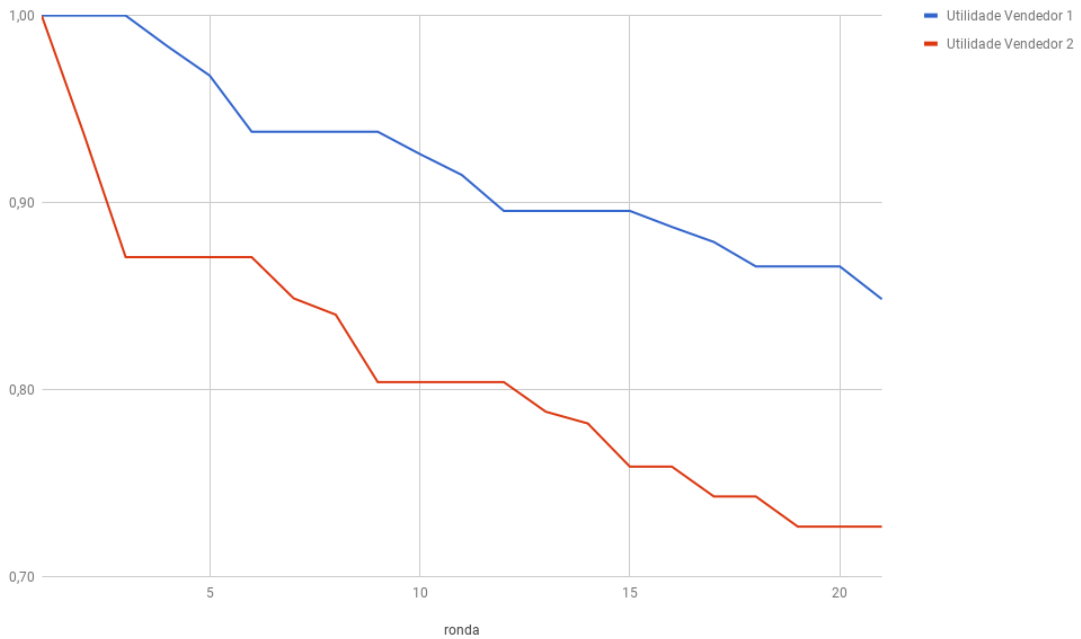


Figura 6.8: Cenário 3: Negociação com argumentação - Utilidade dos vendedores

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Experimentação e resultados

Comprador envia: INFORM_IF BETTER_PROPOSAL DELIVERY 4.045312 para o vendedor 2.

Como o valor mínimo de data de entrega para o vendedor 2 é 4.0, o agente não pode aceitar o apelo do comprador. No entanto, para tentar não perder a negociação, envia uma recompensa, que se traduz numa cedência noutra atributo.

Comprador recebe: INFORM_IF REWARD P:1.0 Q:866822.94 E:4.55664 do vendedor 2, ou seja, não altera o valor da entrega para menos de 4.04 mas recompensa com um preço de 1.0

Como a utilidade da proposta de recompensa (0.926898) é superior à do outro agente competidor, o agente aceita a nova proposta.

Ronda 19

Propostas atuais: Vendedor 1: P:2.048 Q:870140 E:4.045312 com utilidade de 0.91351455
Vendedor 2: P:1.0 Q:866822.94 E:4.55664 com utilidade de 0.926898

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia : INFORM_IF BETTER_PROPOSAL PRICE 1.0 para o vendedor 1

Comprador recebe: INFORM_IF REWARD 2.048 870140 3.0 do vendedor 1.

O vendedor 1 não altera o valor do preço mas recompensa com uma data de entrega de 3.
Como a utilidade da proposta de recompensa (0.93093646) é superior à do outro agente competidor, o agente aceita a nova proposta.

Ronda 20

Propostas atuais: Vendedor 1: P:2.048 Q:870140 E:3.0 com utilidade de 0.93093646
Vendedor 2: 1.0 866822.94 4.55664 com utilidade de 0.926898

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia : INFORM_IF BETTER_PROPOSAL DELIVERY 3.0 para o vendedor 2

Comprador recebe: INFORM_IF REWARD P:1.0 Q:900000.0 E:4.55664 com utilidade de 0.9423399 do vendedor 2. O vendedor não cede mais no atributo data de entrega, mas envia como

Experimentação e resultados

recompensa envia uma quantidade de 900000 de produto.

Como a utilidade da proposta de recompensa (0.9407227) é superior à do outro agente competidor, o agente aceita a nova proposta.

Ronda 21

Propostas atuais: Vendedor 1: P:2.048 Q:870140 E:3.0 com utilidade de 0.93093646

Vendedor 2: P:1.0 Q:900000.0 E:4.55664 com utilidade de 0.9407227

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia : INFORM_IF BETTER_PROPOSAL QUANTITY 900000 para o vendedor 1

Comprador recebe: INFORM_IF REWARD 1.0 870140.7459999999 3.0 com utilidade de 0.9542253 do vendedor 1. O vendedor não cede na quantidade pois já atingiu o seu limite, mas oferece como recompensa um preço de 1. A recompensa é aceite.

Ronda 22

Propostas atuais: Vendedor 1: P:1 Q:870140 E:3.0 com utilidade de 0.9542253

Vendedor 2: P:1.0 Q:900000.0 E:4.55664 com utilidade de 0.9407227

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia : INFORM_IF BETTER_PROPOSAL DELIVERY 3.0 para o vendedor 2

Comprador recebe REFUSAL do vendedor 2, pois este não pode ceder no atributo proposto pelo comprador e não pode enviar recompensas pois já excedeu todo o seu domínio de negociação.

Vendedor 1 ganha a negociação com utilidade para o comprador de 0.9542253.

A utilização também do argumento recompensa permitiu prolongar por mais algumas rondas a negociação e melhorar ligeiramente a utilidade do agente comprador em comparação com o sistema de negociação com argumentação do cenário 2, de 0.91 para 0.954. Em comparação com o sistema de negociação sem argumentação, a utilidade do agente comprador é ultrapassada em apenas 22 rondas.

6.4 Cenário 4

Com este cenário pretende-se analisar o comportamento do sistema quando é permitido todos os tipo de argumentos.

Serão discutidas duas negociações, uma primeira de um carregamento de lenços e outra de um carregamento de *chiffon*.

A primeira negociação é idêntica à negociação do cenário 2, no entanto o *array* de necessidades do agente comprador inclui *chiffon* e o *array* de produtos para vender do vendedor 2 também inclui *chiffon*, o agente vendedor 1 mantém-se inalterado. Na segunda negociação, o vendedor 1 é substituído por um vendedor de *chiffon*, o vendedor 3.

De seguida são especificados os domínios da negociação de *chiffon* para cada agente.

Agente comprador:

- Preço mínimo: 1; Preço máximo: 10; Preço preferido: 2
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 900000
- Data de entrega mínima: 1; Data de entrega máxima: 10; Data de entrega preferida: 3
- Utilidade mínima: 0.5; Peso preço: 0.2, Peso quantidade: 0.3; Peso Entrega: 0.5

Agente vendedor 2:

- Preço mínimo: 2; Preço máximo: 10; Preço preferido: 5
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 700000
- Data de entrega mínima: 3; Data de entrega máxima: 31; Data de entrega preferida: 7
- Utilidade mínima: 0.5; Peso preço: 0.15, Peso quantidade: 0.35; Peso Entrega: 0.5

Agente vendedor 3:

- Preço mínimo: 1; Preço máximo: 10; Preço preferido: 7
- Quantidade mínima: 180000; Quantidade máxima: 900000; Quantidade preferida: 500000
- Data de entrega mínima: 4; Data de entrega máxima: 31; Data de entrega preferida: 9
- Utilidade mínima: 0.5; Peso preço: 0.15, Peso quantidade: 0.35; Peso Entrega: 0.5

6.4.1 Sistema sem argumentação

Na primeira negociação, carregamento de lenços, a negociação ocorre nos mesmos termos da negociação do cenário 2 (secção 6.2.1), sendo escolhida a proposta do vendedor 1 com uma utilidade para o comprador de 0.93.

Experimentação e resultados

Os valores de utilidade para a segunda negociação, negociação de *chiffon*, encontram-se nas figuras 6.9 (utilidades do comprador), e 6.9 (utilidades dos vendedores). Esta negociação terminou ao fim de 25 rondas.

A negociação começa com duas propostas iniciais com utilidade para o comprador de 0.37 e 0.61. De seguida inicia-se o ciclo de *feedback* onde cada agente vai ceder de acordo com o *feedback* do agente comprador. Como se pode observar pelo gráfico 6.9, a utilidade das propostas para o agente vendedor vão aumentando muito ligeiramente até quase se estabilizarem. Isto deve-se ao facto de os agentes vendedores cederem primeiro no atributo que menos afeta a sua função de utilidade e neste cenário, o atributo que menos afeta os agentes vendedores é também o atributo que menos afeta o agente comprador. Mas de seguida existe uma subida repentina no valor de utilidade para o agente comprador, isto deve-se aos agentes começarem a ceder no seu segundo atributo menos preferido, que é o atributo que mais importância tem no valor de utilidade do agente comprador. A negociação terminou na ronda 25, vencendo a negociação o agente vendedor 2 com uma proposta de utilidade de 0.82 para o agente comprador.

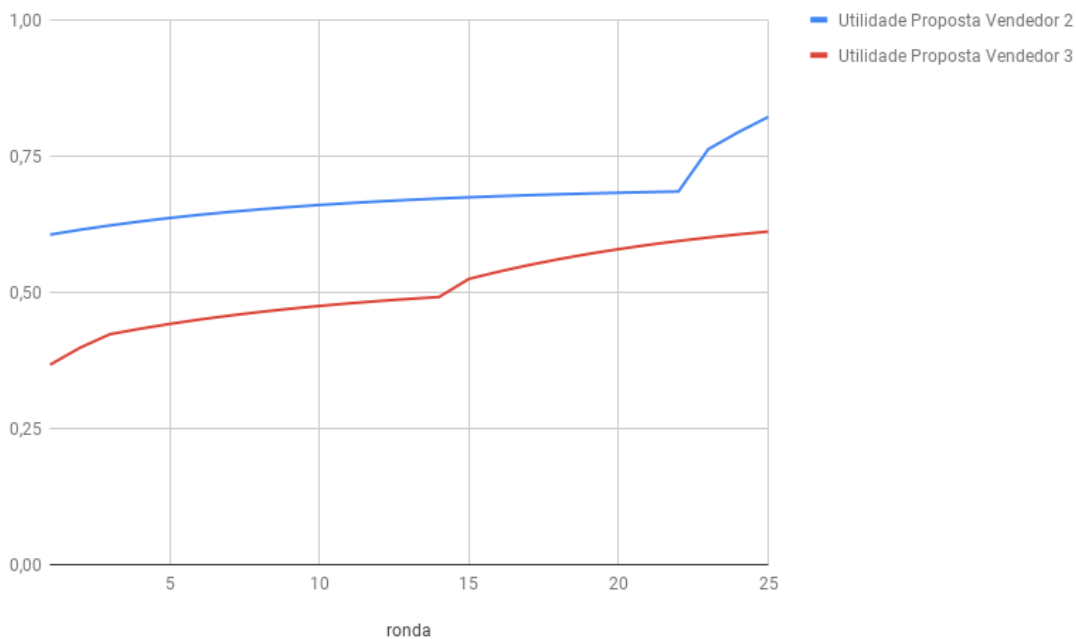


Figura 6.9: Cenário 4: Negociação sem argumentação - Utilidade comprador

6.4.2 Sistema com argumentação

Na primeira negociação, a negociação ocorre nos mesmos termos do cenário 3 (secção 6.3.2) até à ronda 22, com os agentes a trocaram apelos ao interesse próprio e recompensas. Como neste

Experimentação e resultados

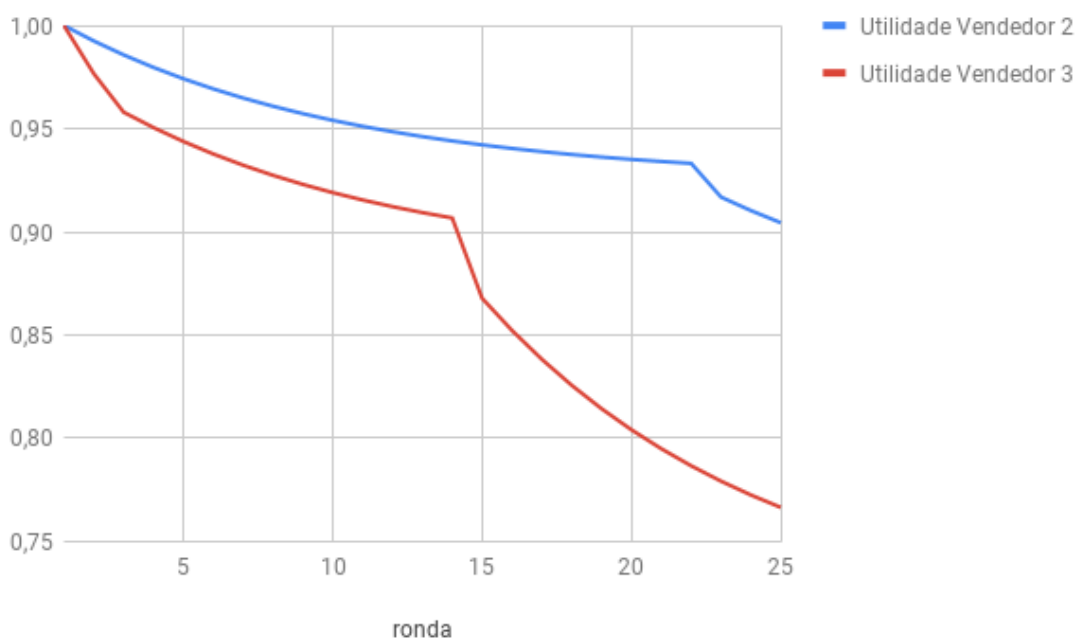


Figura 6.10: Cenário 4: Negociação sem argumentação - Utilidade dos vendedores

cenário é possível utilizar promessas, os agentes tiram partido disso. De seguida são descritas as mensagens trocadas e ações dos agentes na ronda 22.

P representa Preço, Q representa Quantidade e E representa Data de Entrega

Ronda 22

Propostas atuais: Vendedor 1: P:1 Q:870140 E:3.0 com utilidade de 0.9542253

Vendedor 2: P:1.0 Q:900000.0 E:4.55664 com utilidade de 0.9407227

Comprador verifica se existem promessas por cumprir, não existindo, utiliza um apelo ao interesse próprio.

Comprador envia : `INFORM_IF BETTER_PROPOSAL DELIVERY 3.0` para o vendedor 2

Comprador recebe: `INFORM_IF PROMISE CHIFFON DELIVERY 3.0` do vendedor 2. O agente não pode ceder no atributo proposto pelo comprador e não pode enviar recompensas pois já excedeu todo o seu domínio de negociação. No entanto como também vende *chiffon*, e uma data de entrega de 3 está dentro do domínio de negociação de *chiffon*, o agente envia uma promessa onde diz que caso o comprador aceite a sua proposta e finalize a negociação lhe dará uma data de entrega de 3 no próximo carregamento de *chiffon*.

Experimentação e resultados

O agente comprador decide aceitar a promessa e finaliza a negociação, o vendedor 2 ganha a negociação com utilidade de 0.9407227 para o comprador.

Neste cenário, a utilização de promessas, permitiu ao agente que não teve a proposta com maior utilidade ganhar a negociação, comprometendo-se a cumprir a sua promessa no futuro.

Na segunda negociação (negociação de *chiffon*) será possível observar o efeito da promessa feita pelo agente.

Nas figuras 6.11 e 6.12 encontram-se as utilidades do comprador e dos vendedores, respectivamente, na negociação de *chiffon*.

A negociação de *chiffon* no sistema com argumentação leva apenas nove rondas a ser concluída. O agente comprador recebe duas propostas iniciais com utilidades de 0.37 e 0.61.

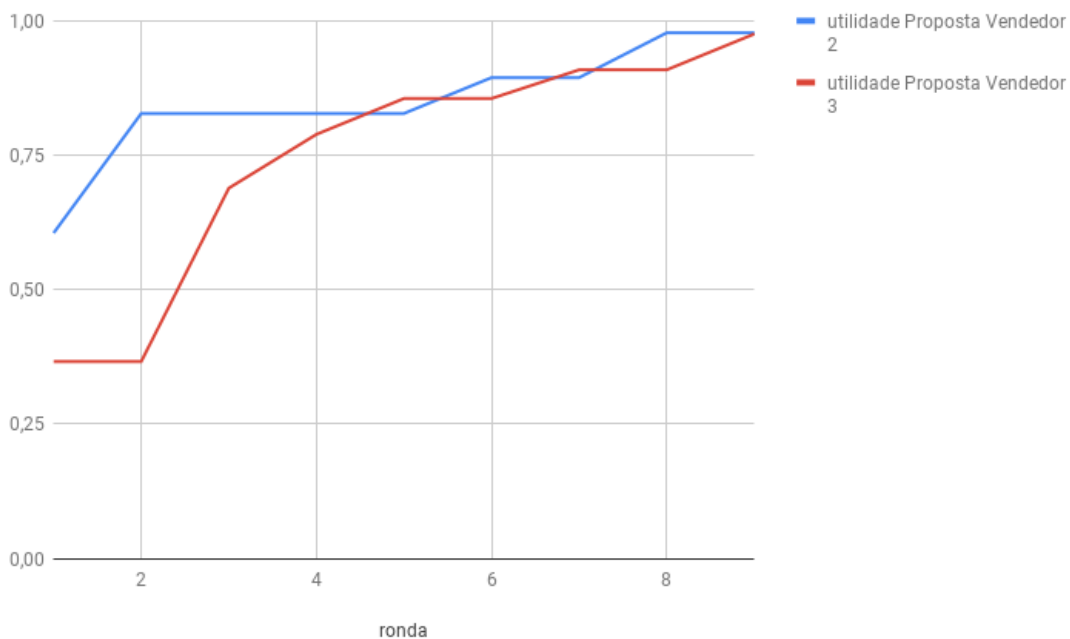


Figura 6.11: Cenário 4: Negociação com argumentação - Utilidade comprador

Nesta negociação os agentes podem utilizar todos os argumentos e é interessante analisar o que acontece na ronda 2. As mensagens trocadas entre os agentes são apresentadas de seguida.

Ronda 2

Propostas atuais: Vendedor 2: P:5.0 Q:700000.0 E:7.0 com utilidade de 0.6055556
Vendedor 3: P:7.0 Q:500000.0 E:9.0 com utilidade 0.36666664

Experimentação e resultados

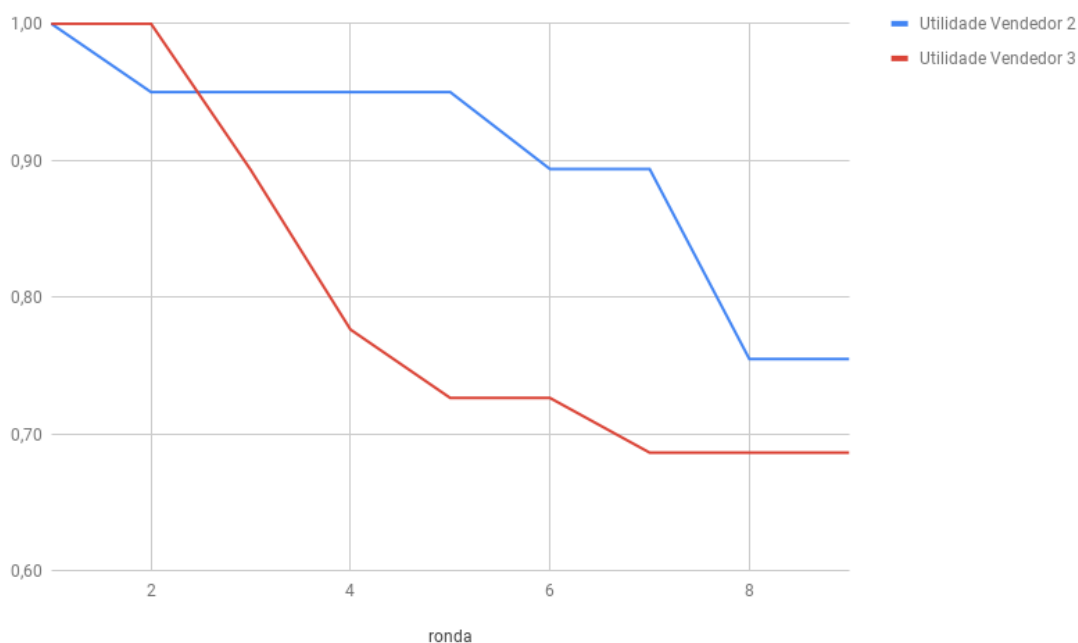


Figura 6.12: Cenário 4: Negociação com argumentação - Utilidade dos vendedores

Comprador verifica se existem promessas por cumprir. Existe uma promessa do vendedor 2 que promete uma data de entrega de 3 em *chiffon*. Comprador envia um apelo a que a promessa seja cumprida.

Comprador envia: `INFORM_IF PROMISE_MADE Chiffon DELIVERY 3`

Agente vendedor verifica se no passado fez essa promessa. Após verificar envia uma nova proposta que cumpre a promessa.

Comprador recebe: `PROPOSE P:5.0 Q:700000.0 E:3.0` com utilidade de 0.8277778 para o comprador.

Todo o resto da negociação ocorre da mesma forma dos cenários anteriores, trocando apelos ao interesse próprio e recompensas. A promessa ao ser cumprida acelerou a negociação. Ao fim de apenas 9 rondas a negociação tem duas propostas com utilidades muito próximas para o agente comprador, de 0.977777 (vendedor 2) e 0.975555 (vendedor 3), sendo escolhida a do vendedor 2 pois o vendedor 3 rejeita ceder mais.

Em comparação com o sistema sem argumentação, a negociação foi muito mais rápida e alcançou valores de utilidades mais elevados sendo que o uso de promessas permitiu influenciar negociações futuras.

6.5 Resumo

Neste capítulo foram apresentados os cenários para teste dos sistemas de negociação com e sem argumentação. Foi realizada uma análise dos resultados obtidos e efetuadas comparações entre os diferentes cenários e uso ou não de argumentação no processo de negociação.

Capítulo 7

Conclusões e Trabalho Futuro

Este capítulo conclui esta dissertação apresentando as contribuições do trabalho desenvolvido e sugerindo possíveis linhas de ação para trabalho futuro.

7.1 Contribuições

No capítulo 1 o objetivo desta dissertação foi definido como a especificação e implementação de um sistema multi-agente que permite negociação automática com argumentação num contexto de negócio eletrónico.

No capítulo 4 foi descrito um sistema base de negociação sem argumentação. Foram especificados os agentes participantes do sistema, compradores e vendedores, ciclos de vida destes e a sua forma de ação, bem como o objeto de negociação, as diferentes estruturas de dados, função de utilidade.

No capítulo 5 foi descrito um sistema de negociação baseado no sistema base 4 que permite aos agentes usar argumentação. Foi definido o conceito de argumentação e enumerados os tipos de argumentos que os agentes podem utilizar. Foi apresentada a forma de comunicação e os tipos de mensagem trocadas entre os agentes, bem como a forma de decisão e ações dos agentes.

No capítulo 6 as experiências e testes baseados em cenários simulados foram descritos e discutidos, mostrando o potencial de sistemas de negociação baseados em argumentação.

Após todas as considerações feitas a contribuição desta dissertação foi a esperada, o desenvolvimento de um sistema de negociação automática baseada em argumentação que permite alcançar acordos benéficos para todos os participantes.

7.2 Trabalho Futuro

Tendo como ponto de partida o trabalho desenvolvido nesta dissertação existem várias possibilidades de trabalho futuro.

Conclusões e Trabalho Futuro

Começando pela melhoria do sistema de argumentação, adicionando novos argumentos, novas formas de contra argumentar, introduzindo a argumentação em diferentes contextos e até utilizando processamento de linguagem natural na argumentação.

Outra linha de ação para trabalho futuro é transformar a negociação de um-para-muitos participantes numa negociação de muitos-para-muitos participantes, ajustando a argumentação a esse cenário. Será interessante comparar os resultados com os obtidos nesta dissertação.

Melhorar os sistemas de cedência, envio de recompensas e promessas, aplicando técnicas de aprendizagem computacional, por exemplo usando uma abordagem de *Case based reasoning*.

Uma possibilidade é também introduzir métricas de confiança e fidelidade na aceitação de promessas.

Desenvolver uma interface gráfica que permita controlar os aspetos da negociação, como cenários, argumentos permitidos e possibilite a visualização gráfica de resultados.

Referências

- [ARL03] Ronald Ashri, Iyad Rahwan e Michael Luck. Architectures for negotiating agents. *Multi-Agent Systems and Applications III: Proceedings of the 3rd International Central and Eastern European Conference on Multi-Agent Systems (CEEMAS 2003)*, volume 2691 of *Lecture Notes in Artificial Intelligence*, 2691:136–146, 2003.
- [BPR99] Fabio Bellifemine, Agostino Poggi e Giovanni Rimassa. JADE—A FIPA-compliant agent framework. *Proceedings of PAAM*, pages 97–108, 1999.
- [DJH⁺13] X.-J. Dong, C Jian, Y.-L. Hu, G.-R. Jiang e T.-Y. Huang. Multi-attribute negotiation model based on internal factors argumentation. *International Conference on Management Science and Engineering - Annual Conference Proceedings*, pages 20–27, 2013.
- [ESM12] Ashraf B. El-Sisi e Hamdy M. Mousa. Argumentation based negotiation in multiagent system. *Proceedings - ICCES 2012: 2012 International Conference on Computer Engineering and Systems*, 3(3):261–266, 2012.
- [FP97] Foundation For e Intelligent Physical. Agent Communication Language. *FIPA 97 Specification Foundation for Intelligent Physical Agents*, 5(6), 1997.
- [FRN09] Angelika Först, Achim Rettinger e Matthias Nickles. Argumentation- vs. proposal-based negotiation: An empirical case study on the basis of game-theoretic solution concepts. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5384 LNAI:161–180, 2009.
- [Gru93] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.
- [JAD] JADE Architecture Overview. Acedido em 20-06-2018. URL: <http://jade.tilab.com/doc/tutorials/JADEAdmin/jadeArchitecture.html>.
- [JPNS] N R Jennings, S Parsons, P Noriega e C Sierra. On Argumentation-Based Negotiation. pages 1–8.
- [KSE98] Sarit Kraus, Katia Sycara e Amir Evenchik. Reaching agreements through argumentation: a logical model and implementation. *Artificial Intelligence*, 104(1-2):1–69, 1998.
- [Mae95] Pattie Maes. Artificial life meets entertainment: lifelike autonomous agents. *Communications of the ACM*, 38(11):108–114, 1995.
- [Nog02] Miguel Pinho Nogueira. *Sistema Multi-Agente para Mediação Electrónica de Seguros*. Master’s dissertation, 2002.

REFERÊNCIAS

- [OR01] Eugénio Oliveira e Ana Rocha. Agents Advanced Features for Negotiation in Electronic Commerce and Virtual Organisations Formation Process. *Agent Mediated Electronic Commerce*, 1991:78–97, 2001.
- [RN10] Stuart J Russel e Peter Norvig. *Artificial Intelligence a Modern Approach*. 2010.
- [ZJH10] Ge Zhang, Guo Rui Jiang e Ti Yun Huang. Design of argumentation-based multi-agent negotiation system oriented to E-commerce. *International Conference on Internet Technology and Applications, ITAP 2010 - Proceedings*, pages 0–5, 2010.
- [ZL10] Liang Zhang e Na Li. Multi-agent negotiation system in online auction. *2010 2nd International Conference on Communication Systems, Networks and Applications, ICCSNA 2010*, 2:81–84, 2010.