

**UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ADMINISTRAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ADMINISTRAÇÃO**

**AGENTES DE SOFTWARE NAS EMPRESAS: UM ESTUDO DE CASO
NA ÁREA DE COMPRAS**

EDMILSON LUCENA NÉRI

PORTO ALEGRE, 2002.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ADMINISTRAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ADMINISTRAÇÃO
DISSERTAÇÃO DE MESTRADO

AGENTES DE SOFTWARE NAS EMPRESAS: UM ESTUDO DE CASO NA
ÁREA DE COMPRAS

Edmilson Lucena Néri

Dissertação de Mestrado, apresentada ao Programa de Pós-Graduação em Administração da Universidade Federal do Rio Grande do Sul como requisito para a obtenção do título de Mestre em Administração.

Orientador: Prof. Dr. Norberto Hoppen

Porto Alegre, 2002.

Aos meus pais: Vera e Edson Néri.

Agradecimentos

Agradeço ao CNPq pela bolsa concedida, à UFRGS e ao PPGA pela oportunidade de ter realizado este curso de mestrado.

Agradeço ao Prof. Dr. Norberto Hoppen pela orientação, atenção, compreensão, confiança e oportunidades oferecidas. Também aos professores da banca: Prof. Dr. Carlos Cano, Prof. Dr. Wagner Kaehler e ao Prof. Dr. Eduardo Santos. Ao Prof. Dr. Denis Borenstein pelas inúmeras dicas dadas durante todo o curso.

À equipe de Obtenção de TI da Brasil Telecom, em especial a Henrique Ribeiro, pela atenção e paciência.

Aos amigos de Brasília, em especial Emerson Gomes e Fabiana Fernandes, pelos apoios técnicos e incentivos.

Aos amigos Fábio Dahia e Márcio Dahia pelas discussões e ajudas fundamentais na construção dos mecanismos de decisão e negociação.

Aos amigos e amigas de curso, em especial a André Ferreira e a Sílvia Piovesan, pelo companheirismo, o que foi essencial para manter uma boa saúde mental, principalmente quando se está longe de casa.

Aos meus irmãos Edson, Eveline e Emanuel Lucena Néri, pelos incentivos morais e, finalmente, aos meus pais, Edson da Silva Néri e Vera Lúcia de Lucena Néri, pelos inúmeros incentivos morais e financeiros.

“A vida é uma seqüência natural dos acontecimentos”

Tio Carlos Alberto de Lucena

(em um momento de embriaguez)

RESUMO

Com o aumento de informações disponíveis e das possibilidades de fazer comércio por meios eletrônicos, aumenta a complexidade de algumas tarefas, como comprar. Um dos possíveis caminhos para diminuir essa complexidade é dotar programas de computadores com conhecimentos sobre a realização de tais tarefas, incluindo como tomar decisões. Esta dissertação versa sobre a utilização de agentes de software nas empresas, que são programas de computadores delegados para a realização de funções específicas. Além de uma revisão sobre o atual uso de tais tecnologias, apresenta-se uma possível aplicação na área de Compras para uma empresa real. Para modelar o sistema proposto, utilizou-se uma extensão da *Unified Modeling Language* (UML), a *Agent UML*, que objetiva a adequação da primeira para representar sistemas multiagentes. As dificuldades encontradas para modelagem são uma discussão adicional, bem como as sugestões. Outras contribuições do presente trabalho são as apresentações de mecanismos de decisão e de negociação para a Função Compras, inspirados em funções de valor e na negociação colaborativa. Encerra-se, apontando algumas futuras direções, incentivando reflexões posteriores sobre este tema pioneiro na Escola de Administração da Universidade Federal do Rio Grande do Sul.

ABSTRACT

With the increasing amount of information and possibilities available to make commerce throughout electronic ways nowadays, some tasks, like purchaser, became more complex. One of the possible ways to deal with this complexity is endowing computer programs with knowledge over these tasks, including how to make decisions. This thesis is about the use of software agents within organizations. Software agents could be understood as computer programs delegated to specific functions. Beyond software agents' review, it will be presented a possible application for software agents in a purchaser department of a real organization. To model the proposed system it was used a Unified Modeling Language's extension called Agent UML that has a goal to fit the former to agent based systems modeling. The found difficulties modeling are an additional matter, as well as the suggestions. Some other contributions of the current work are the decision and negotiation mechanisms applied to the purchaser function. They were inspired on value functions and on integrative negotiation. There are some future works suggestions in the end, as well as a discussion about the insights raised from the whole work, which has a pioneer approach in this school.

SUMÁRIO

RESUMO	5
ABSTRACT.....	6
LISTA DE TABELAS	11
1 INTRODUÇÃO	12
1.1 OBJETIVOS	13
1.2 OS CAPÍTULOS	13
2 AGENTES	15
2.1 SISTEMAS MULTIAGENTES	16
2.2 APLICAÇÕES DE AGENTES DE SOFTWARE.....	17
2.3 APLICAÇÕES NO COMÉRCIO	18
2.4 SISTEMAS MULTIAGENTES PARA O COMÉRCIO	20
2.5 CUIDADOS PARA O COMÉRCIO ELETRÔNICO	22
2.6 MODELANDO AGENTES	24
2.6.1 <i>A Metodologia Gaia</i>	24
2.6.2 <i>Agent UML</i>	27
3 COMPRAS, DECISÃO E NEGOCIAÇÃO	30
3.1 COMPRAS.....	30
3.1.1 <i>A importância da Função Compras para a organização</i>	30
3.1.2 <i>A Função Compras na Administração da Produção</i>	30
3.1.3 <i>Tipos de itens de compra</i>	31
3.1.4 <i>Variáveis-chave de compras</i>	32
3.2 ANÁLISE DE DECISÃO.....	33
3.2.1 <i>Um panorama do processo de análise de decisão</i>	33
3.2.2 <i>Paradigma de Análise de Decisão segundo Keeney e Raiffa (1976)</i>	35
3.2.3 <i>Teoria da Utilidade Esperada</i>	36
3.2.4 <i>Axiomatização</i>	36
3.2.5 <i>Diferença entre função de valor e função de utilidade</i>	37
3.2.6 <i>Alternativas com múltiplos atributos</i>	37
3.2.7 <i>Dominância</i>	38
3.2.8 <i>Fronteira eficiente</i>	38
3.2.9 <i>Curvas de indiferença</i>	38
3.2.10 <i>Paradoxos</i>	39
3.3 NEGOCIAÇÃO.....	39

3.3.1	<i>Negociação em compras</i>	39
3.3.2	<i>Negociação integrativa</i>	40
3.3.3	<i>Computadores e negociação</i>	42
4	UMA OPORTUNIDADE PARA USO DE AGENTES	43
4.1	O CENÁRIO.....	43
4.2	UMA SOLUÇÃO ALTERNATIVA PARA A COMPRA DE SUPRIMENTOS DE TI.....	47
4.2.1	<i>Especificação do Sistema Multiagente para Compra de Suprimentos de TI</i>	48
4.2.2	<i>Limitações</i>	50
4.3	UMA SÍNTESE DOS PROCESSOS ATUAL E PROPOSTO.....	51
4.3.1	<i>Atual</i>	51
4.3.2	<i>Proposto</i>	51
5	MODELANDO O SISTEMA MULTIAGENTE COM AUML	52
5.1	DIAGRAMA DE CONTEXTO.....	52
5.2	DIAGRAMAS DE CASOS DE USO.....	53
5.3	ATRIBUTOS DOS AGENTES.....	61
5.4	CONJUNTO DE MENSAGENS.....	63
5.5	DIAGRAMAS DE INTERAÇÃO.....	66
5.6	DIAGRAMAS DE ATIVIDADES.....	71
6	MECANISMOS DE DECISÃO E NEGOCIAÇÃO	75
6.1	CONSTRUINDO AS FUNÇÕES DE VALOR.....	76
6.1.1	<i>Os limites das funções dos negociadores</i>	80
6.2	O ALGORITMO PARA NEGOCIAÇÃO INTEGRATIVA.....	81
6.3	A IMPLEMENTAÇÃO.....	84
6.4	OS RESULTADOS.....	85
6.4.1	<i>Cenário de Testes 1</i>	86
6.4.2	<i>Cenário de Testes 2</i>	87
6.4.3	<i>Cenário de Testes 3</i>	89
6.4.4	<i>Teste 4</i>	90
6.5	DISCUSSÃO DOS MECANISMOS.....	92
7	CONSIDERAÇÕES FINAIS	93
7.1	RESULTADOS ALCANÇADOS.....	93
7.1.1	<i>Modelagem</i>	93
7.1.2	<i>Mecanismos de Decisão e Negociação</i>	94
7.1.3	<i>Comentários Finais</i>	94
7.2	LIMITAÇÕES DO ESTUDO.....	94
7.2.1	<i>Estudo de Caso</i>	94
7.2.2	<i>Modelagem</i>	95

7.2.3	<i>Mecanismos de Decisão e Negociação</i>	95
7.3	RECOMENDAÇÕES PARA ESTUDOS FUTUROS.....	95
8	REFERÊNCIAS BIBLIOGRÁFICAS	97
	APÊNDICE A – POSSÍVEL TABELA PARA FORMATAÇÃO DE UM CONTRATO ABERTO	103

LISTA DE FIGURAS

FIGURA 1 - UMA TAXONOMIA PARA AGENTES AUTÔNOMOS (FRANKLIN E GRAESSER, 1996)	16
FIGURA 2 - CONCEITOS DE ANÁLISE DA GAIA	26
FIGURA 3 - RELACIONAMENTO ENTRE OS MODELOS DA GAIA	27
FIGURA 4 - FUNÇÃO COMPRAS (SLACK ET AL., 1997)	31
FIGURA 5 - PROCESSO DECISÓRIO (CLEMEN, 1996)	34
FIGURA 6 - CURVAS DE INDIFERENÇA E CURVA DE CONTRATO.....	41
FIGURA 7 - DIAGRAMA DE CONTEXTO.....	53
FIGURA 8 - DIAGRAMA DE CONTEXTO EXPLODIDO EM ATORES.....	55
FIGURA 9 - DIAGRAMA DE CASOS DE USO (EXPLOSÃO DOS AGENTES COMPRADORES).....	56
FIGURA 10 - DIAGRAMA DE CASOS DE USO (EXPLOSÃO DOS AGENTES FORNECEDORES).....	58
FIGURA 11 - DIAGRAMA DE CASOS DE USO (EXPLOSÃO DO AGENTE MERCADO)	60
FIGURA 12 - DIAGRAMA DE SEQÜÊNCIA: NÚCLEOS E AG. COMPRADORES.....	66
FIGURA 13- DIAGRAMA DE SEQÜÊNCIA: FORNECEDORES E AG. FORNECEDORES.....	68
FIGURA 14- DIAGRAMA DE SEQÜÊNCIA: DMS E AG. MERCADO.....	69
FIGURA 15- DIAGRAMA DE SEQÜÊNCIA: AG. FORNECEDORES E AG. MERCADO.....	69
FIGURA 16- DIAGRAMA DE SEQÜÊNCIA: AG. COMPRADOR E AG. MERCADO.....	70
FIGURA 17 - DIAGRAMA DE SEQÜÊNCIA: AG. COMPRADORES E AG. FORNECEDORES.....	70
FIGURA 18 - DIAGRAMA DE ATIVIDADES DE UM AG. COMPRADOR.....	72
FIGURA 19 - DIAGRAMA DE ATIVIDADES DE UM AG. FORNECEDOR.....	73
FIGURA 20 - DIAGRAMA DE ATIVIDADES DO AG. MERCADO (REGISTRANDO AGENTES)	74
FIGURA 21 - EXEMPLOS DAS FUNÇÕES EXPONENCIAIS E INVERSAS.....	79
FIGURA 22 - IMPLEMENTAÇÃO DO ALGORITMO DE NEGOCIAÇÃO.....	82
FIGURA 23 - IMPLEMENTAÇÃO DO MÉTODO ENVIAPROPOSTA().....	83

LISTA DE TABELAS

TABELA 1 - CASO DE USO: MONTA PEDIDO.....	56
TABELA 2 - CASO DE USO: INFORMA STATUS.....	57
TABELA 3 - CASO DE USO: INFORMA RESULTADO DA COMPRA.....	57
TABELA 4 - CASO DE USO: ENVIA PEDIDOS.....	57
TABELA 5 - CASO DE USO: NEGOCIA MELHORIA DO ACORDO.....	57
TABELA 6 - CASO DE USO: ESCOLHE COTAÇÃO.....	58
TABELA 7 - CASO DE USO: MONTA VENDA.....	59
TABELA 8 - CASO DE USO: INFORMA STATUS DAS VENDAS.....	59
TABELA 9 - CASO DE USO: ENVIA COTAÇÃO.....	59
TABELA 10 - CASO DE USO: NEGOCIA MELHORIAS.....	59
TABELA 11 - CASO DE USO: MANTER LISTAS E TABELAS.....	60
TABELA 12 - CASO DE USO: CONTROLAR PARTICIPAÇÃO DOS AGENTES.....	60
TABELA 13 - CASO DE USO: INFORMAR DETALHES DAS COMPRAS.....	61
TABELA 14 - CASO DE USO: INFORMAR HISTÓRICO.....	61
TABELA 15 - PRINCIPAIS ATRIBUTOS DO AGENTE MERCADO.....	61
TABELA 16 - PRINCIPAIS ATRIBUTOS DO AGENTE COMPRADOR.....	62
TABELA 17 - PRINCIPAIS ATRIBUTOS DO AGENTE FORNECEDOR.....	62
TABELA 18 - CONJUNDO DE MENSAGENS TROCADAS ENTRE OS ATORES.....	63
TABELA 19 - CONFIGURAÇÃO DO COMPRADOR NO CENÁRIO DE TESTES 1.....	86
TABELA 20 - CONFIGURAÇÃO E AVALIAÇÃO DOS FORNECEDORES NO CENÁRIO DE TESTES 1.....	86
TABELA 21 - COMPARATIVO ANTES/APÓS NEGOCIAÇÃO NO CENÁRIO DE TESTES 1.....	87
TABELA 22 - CONFIGURAÇÃO DO COMPRADOR NO CENÁRIO DE TESTES 2.....	88
TABELA 23 - CONFIGURAÇÃO E AVALIAÇÃO DOS FORNECEDORES NO CENÁRIO DE TESTES 2.....	88
TABELA 24 - COMPARATIVO ANTES/APÓS NEGOCIAÇÃO NO CENÁRIO DE TESTES 2.....	89
TABELA 25 - CONFIGURAÇÃO DO COMPRADOR NO CENÁRIO DE TESTES 3.....	89
TABELA 26 - CONFIGURAÇÃO E AVALIAÇÃO DOS FORNECEDORES NO CENÁRIO DE TESTES 3.....	90
TABELA 27 - COMPARATIVO ANTES/APÓS NEGOCIAÇÃO NO CENÁRIO DE TESTES 3.....	90
TABELA 28 - CONFIGURAÇÃO DO COMPRADOR NO CENÁRIO DE TESTES 4.....	91
TABELA 29 - CONFIGURAÇÃO E AVALIAÇÃO DOS FORNECEDORES NO CENÁRIO DE TESTES 4.....	91
TABELA 30 - COMPARATIVO ANTES/APÓS NEGOCIAÇÃO NO CENÁRIO DE TESTES 4.....	92

1 INTRODUÇÃO

Já há algum tempo, a automação vem tornando-se mais presente nas empresas. Fábricas diminuíram drasticamente o número de funcionários na linha de produção, substituindo-os por máquinas, que trabalham com mais precisão, rapidez e custam menos, conseqüentemente, aumentam a produtividade. A maior parte das tarefas automatizadas, entretanto, tem caráter mecânico, ou, em outras palavras, não são trabalhos intelectuais.

Uma outra vertente dos avanços tecnológicos é o aumento de informações armazenadas em meios digitais e disponíveis através de sistemas de informação que operam em redes, em especial, a Internet. O aumento foi (e continua sendo) tão grande, que se tornou excessivo. Apesar de se ter até criado novas necessidades de informação, houve o surgimento de inúmeras informações não necessárias.

Com as novas tecnologias, possibilidades novas também surgiram, inclusive para os negócios. Com as redes de comunicação mundiais para troca de informações, as distâncias e o tempo somem para a negociação. Potencialmente, parceiros multiplicam-se de forma antes inimagináveis. Ainda há barreiras geográficas e temporais, evidentemente, quando os bens ou produtos negociáveis são concretos, ou, mais bem explicitado, necessitam ser transportados entre os parceiros, após negociação.

Vê-se que a produtividade, as informações e os parceiros de negociação crescem com a tecnologia, mas também aumenta a complexidade em lidar com tamanhos números. Nas empresas, nem sempre, o simples aumento de funcionários implica conseguir lidar com esta complexidade. A solução deve também ser o uso de tecnologia. A automação deve deixar de ser um conceito associado à substituição apenas do trabalho manual, generalizando-se para a substituição do trabalho, seja manual ou intelectual.

Uma das formas possíveis de automatizar tarefas que exijam conhecimento e tomada de decisão é o uso de agentes de software, que, em uma visão mais ampla possível, são

programas de computadores para os quais tarefas são delegadas. Estes agentes devem ter conhecimentos embutidos para a realização das tarefas, dentre eles, mecanismos de decisão, pois senão não se constituiria uma delegação, já que não haveria uma transferência do poder de decidir algo.

1.1 OBJETIVOS

Dado todo o cenário acima, chega-se aos objetivos da dissertação:

- ◆ Discutir sobre agentes de software, seus usos atuais e possíveis nas empresas, provocando uma reflexão sobre os impactos causados, quando, ou se, adotados;
- ◆ Analisar, através de um estudo de caso, uma possível aplicação de agentes de software em uma empresa. Em particular, na área de compras de suprimentos de tecnologia da informação (TI) da Brasil Telecom S.A., uma das maiores empresas de telecomunicações brasileira;
- ◆ Modelar um sistema multiagente para a aplicação acima, através da *Agent UML*, que é um conjunto de proposta para extensão da UML;
- ◆ Propor mecanismos de decisão e negociação para o sistema. Em especial, deseja-se analisar a aplicação de funções de valor no processo decisório de compras, bem como aplicar as idéias de Raiffa (1982) sobre negociação integrativa.

1.2 OS CAPÍTULOS

No Capítulo 2, vê-se uma revisão da literatura de agentes de software. Conceitos, aplicações desta tecnologia nas empresas e formas de modelagem são discutidos. Um dos principais enfoques dados no capítulo é o uso de agentes no comércio.

Para modelar um sistema multiagente para compras, necessita-se de alguns conhecimentos preliminares, como, evidentemente, de compras, processo decisório e de negociação. O Capítulo 3 compila estudos feitos sobre estes temas.

O Capítulo 4 apresenta um exemplo real de como são feitas as compras de itens de manutenção de TI na Brasil Telecom S.A. e sugere uma solução alternativa, utilizando agentes de software. O capítulo termina com a especificação dos requisitos para um sistema multiagente, feito com o próprio pessoal de Compras da empresa.

O mais extenso dos capítulos é o Capítulo 5. Ele traz a modelagem do sistema, feita em *Agent UML*, que, na verdade, é um conjunto de proposições para extensão da UML para agentes. Algumas avaliações e sugestões são feitas para aquela linguagem.

O Capítulo 6 apresenta os mecanismos de decisão e negociação do sistema, oferecendo um rico ambiente para debates, sendo mais uma contribuição do presente trabalho.

Considerações finais são apresentadas no Capítulo 7, dentre elas, algumas conclusões, as limitações da atual pesquisa e direções futuras.

No Capítulo 8, encontram-se as referências bibliográficas que inspiraram e nortearam a dissertação.

2 AGENTES

Não há consenso sobre o conceito de agentes. Um dos maiores problemas que os pesquisadores de agentes têm é justamente o conceito de agentes. Vê-se na literatura várias definições, cada uma tentando adequar-se aos propósitos daqueles que as definem.

Em Turban et al. (2000, p.497) há uma coletânea de conceitos. Para citar alguns:

Um agente é alguma coisa que pode ser vista como percebendo seu ambiente através de sensores e que age neste ambiente através de seus atuadores (RUSSEL e NORVIG apud TURBAN et al., 2000).

Um agente inteligente é um software que assiste a pessoas e age em seus lugares. Agentes inteligentes permitem que pessoas deleguem trabalhos que poderiam ser feitos por agentes de software. Agentes podem, assim como assistentes podem, automatizar tarefas repetitivas, lembrar coisas que você esqueceu, resumir inteligentemente dados complexos, aprender por você e até fazer recomendações para você (GILBERT apud TURBAN et al., 2000).

Uma boa discussão sobre o conceito de agentes pode ser encontrada em Franklin e Graesser (1996). Lá se encontra uma outra coletânea de conceitos, com as devidas críticas, bem como, o conceito dado pelos próprios autores, que é:

Um agente autônomo é um sistema situado dentro e uma parte de um ambiente que sente este ambiente e age sobre ele, através do tempo, realizando sua própria agenda e assim afetando o que ele sentirá no futuro (FRANKLIN e GRAESSER, 1996).

Entretanto, os próprios autores da última definição falam que ela é muito ampla. Afirmam que, por exemplo, de seres humanos a um termostato, há uma enorme quantidade de entidades que se encaixam na definição. Para definir subtipos de agentes, os autores definiram propriedades para diferenciá-los.

Assim, eles sugeriram uma taxonomia de agentes inspiradas na taxonomia dos seres vivos, como pode ser observado na Figura 1.

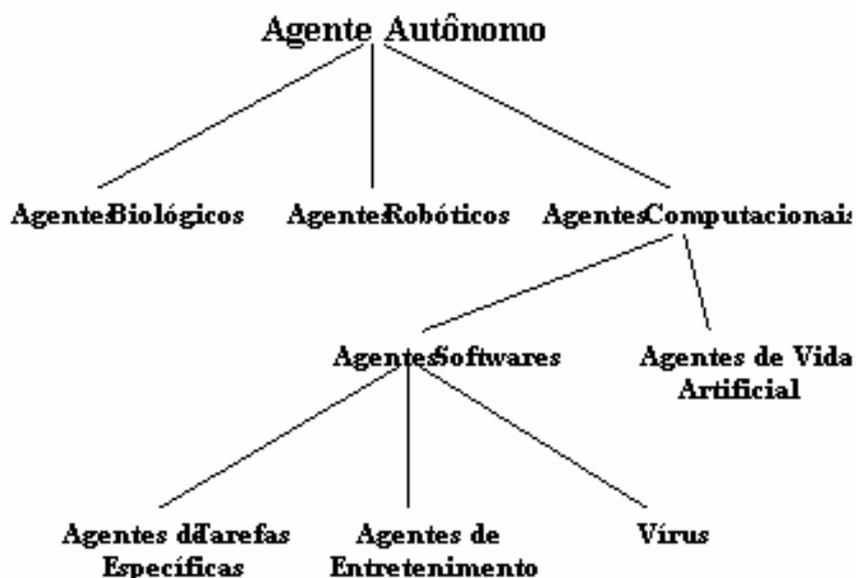


Figura 1 - Uma taxonomia para agentes autônomos (FRANKLIN e GRAESSER, 1996)

Mencionou-se o conceito “agente autônomo” sem haver distinção ao conceito mais simples: agente. Para muitos, o termo “autônomo” é redundante, pois, definem autonomia como uma das propriedades fundamentais de uma entidade para ser considerada um agente.

Deve-se manter em mente que um agente não necessariamente é um software. Liang e Huang (2000), por exemplo, definem agente como sendo uma pessoa ou um negócio autorizado para agir no lugar de outra. Já em desenvolvimento de software, um agente é um programa de computador que pode operar autonomamente e efetuar tarefas singulares sem a direta supervisão humana (HOFFMAN e NOVAK apud LIANG e HUANG, 2000).

2.1 SISTEMAS MULTIAGENTES

Uma propriedade também importante de um agente é a sua comunicação com outros agentes. Há até quem defina, estreitamente, agentes tendo a comunicação como pré-requisito. Por exemplo, agentes de software são “componentes de software que se comunicam com seus colegas através de uma expressiva linguagem de comunicação de agentes” (GENESERETH, 1994).

Tal definição evidencia que a abordagem de agentes é muitas vezes pensada como mais de um agente comunicando-se e colaborando entre si.

Sistemas multiagentes, conforme Stone e Veloso apud Giese (1998), são o subcampo da Inteligência Artificial que proporciona os princípios para construção de sistemas complexos, envolvendo múltiplos agentes e mecanismos para coordenação do comportamento inteligente de um conjunto de agentes.

A última definição tem um viés claro da Inteligência Artificial, mas sistema multiagente pode ser entendido como um sistema que envolve mais de um agente autônomo de forma que estes cooperam entre si para que o sistema consiga atingir os seus objetivos. Para que possam cooperar entre si, necessita haver comunicação.

Para facilitar a interoperabilidade entre os agentes, recomenda-se a utilização de linguagens padrão de comunicação. Um exemplo de uma destas linguagens é a ACL (*Agent Communication Language*), que é composta por três componentes: seu vocabulário, uma linguagem anterior chamada KIF (*Knowledge Interchange Format*) e uma outra linguagem chamada KQML (*Knowledge Query and Manipulation Language*). Uma mensagem ACL é uma expressão KQML em que os argumentos são termos ou sentenças em KIF formadas por palavras do vocabulário ACL (GENESERETH, 1994).

A modelagem apresentada no Capítulo 5 traz um conjunto de mensagens necessárias para a comunicação do agente. Como estas mensagens irão ser traduzidas para a linguagem de comunicação de agentes, depende da escolha do ambiente de implementação.

2.2 APLICAÇÕES DE AGENTES DE SOFTWARE

Muitas são as possíveis aplicações para agentes de software. Quando se têm tarefas passíveis de automação, de natureza distribuída, que exigem alta comunicação entre as partes, que, normalmente, têm diversas especializações; então, muitas vezes, o uso de agentes é uma boa solução computacional para o problema.

Para citar algumas dessas aplicações, tem-se, por exemplo, o uso de agentes para auxiliar a leitura de *e-mails* (ABU-HAKIMA, MCFARLAND e MEECH, 2001). Um sistema multiagente, composto por quatro tipos de agentes, que destacam os principais pontos de um *e-mail* recebido. Adicionalmente, os destaques podem ser enviados via um equipamento de comunicação sem fio, como celulares e *paggers*, para os usuários do sistema.

Um outro exemplo interessante – agora no meio médico – é o uso de um sistema multiagente para ajudar na “mineração” de informações e para anotações sobre o seqüenciamento dos genes dos vírus do herpes. Consiste em três tipos de agentes: agente de extração de informações, agentes de tarefas e agentes de interface. O primeiro, é responsável pela busca de bases de dados sobre os vírus e monitoração das mudanças nas mesmas. O segundo tipo serve para realizar tarefas específicas, como filtragem, análise, integração, etc. O terceiro, realiza a interface com o usuário final do sistema. O sistema multiagente já está em uso e vem trazendo bons resultados, facilitando o trabalho dos pesquisadores dos vírus. A busca das informações já está satisfatória e trabalha-se, agora, na melhoria dos mecanismos de análises e da interface com os usuários (DECKER, ZHENG e SCHMIDT, 2001).

Outros tantos exemplos poderiam ser achados na literatura, mas o enfoque deste trabalho é mostrar a utilização da tecnologia de agentes para aplicações em problemas empresariais, em especial, no comércio.

2.3 APLICAÇÕES NO COMÉRCIO

Agentes de software podem ser aplicados a todas as áreas onde um comportamento personalizado, que execute continuamente e seja semi-autônomo seja desejado. Quanto mais tempo e dinheiro possam ser salvos pela automação, quanto menos riscos estejam envolvidos por se ter uma solução subótima e maior seja a importância de não perder oportunidades, mais apropriado torna-se o uso de agentes (PATTIE, GUTTMAN e MOUKAS, 1999).

Muitos esforços foram e vêm sendo feitos na aplicação de agentes no comércio eletrônico. Seguindo o Modelo CBB (*Consumer Buying Behavior*), pode-se encontrar, em quase todos seus estágios, espaço para a utilização de agentes (PATTIE, GUTTMAN e MOUKAS, 1999). Abaixo, há exemplos do uso de agentes para os estágios de identificação de necessidade, busca de produto, busca por fornecedor e negociação. As etapas de compra: entrega, serviço e avaliação, não estão contempladas nesta revisão.

Em Hernandez e Sen (2000), chama-se a atenção para a incapacidade dos consumidores varejistas estarem atualizados sobre as novidades e tendências de mercado. Sugeriram um agente de software que ajuda o consumidor na escolha de uma compra, através de duas funções básicas: seleção dos produtos e sugestão de produtos alternativos. Na primeira função, o agente verifica se as restrições informadas para a escolha de um produto são factíveis. O agente retorna alguns produtos que poderiam satisfazer as necessidades do

consumidor, caso o mesmo relaxasse em algumas das suas restrições. Na segunda função, de acordo com o desejo do consumidor, o agente pode informar produtos alternativos. Por exemplo, se o consumidor deseja presentear um amigo, que gosta de escutar música em um aparelho móvel, com um *CD-Player* portátil, o agente pode sugerir um *MP3-Player* portátil, como alternativa àquele produto. A arquitetura do agente apresentado tem cinco principais módulos: interface, motor de recuperação de dados, gerador de restrições de mercado, “cruzador” de restrições e um módulo de análise.

Na mesma linha, Dutta, Debnath e Sen (2001), centram as atenções para os problemas dos consumidores ao tentar comprar algo por meio eletrônico. Em geral, uma busca por um produto leva a um retorno de muitas ofertas – ficando difícil analisar todas – de nenhuma ou poucas. A quantidade retornada dificilmente é satisfatória. A sugestão do artigo é a construção de agentes compradores que auxiliam os consumidores, através de reformulações de comandos de busca, quando:

- a) não há retorno. As restrições da busca, por exemplo, podem ter sido não factíveis, como no caso de se desejar um produto de alta qualidade com um preço muito inferior aos ofertados pelo mercado. O agente pode sugerir um produto similar, de qualidade inferior, ao preço desejado;
- b) há muitos retornos. O agente solicita ao usuário que refine sua busca, através da especificação de subcategorias.

Os agentes também têm a funcionalidade de sugerir produtos alternativos, mesmo quando o número de retornos de um comando de busca parece razoável.

Já entrando na etapa de “negociação” do modelo CBB, Liang e Huang (2000) sugerem um *framework* para a aplicação de agentes no suporte ao comércio eletrônico. Eles identificaram seis tipos de comércio, divididos em dois grupos: bilaterais e trilaterais. No grupo dos bilaterais existem dois envolvidos: o vendedor e o comprador. Nos trilaterais ainda há um terceiro, o corretor (*broker*). No primeiro grupo, os mais comuns são: permuta (*barter*), barganha (*bargain*), licitação (*bidding*) e contrato (*contract*). No segundo tipo, os mais comuns são: leilão (*auction*) e compensação (*clearing*).

No seu *framework*, os autores estruturam um sistema multiagente em que cada um dos tipos tem um agente associado. Há um agente chamado “*Market Maker*” que escolhe, de

acordo com as requisições do usuário, qual dos tipos de comércio supracitados é o adequado para a comercialização de determinado produto.

Ainda com foco no segmento *business-to-consumer*, com a preocupação de munir com agentes apenas os consumidores, Preist, Byde e Bartolini (2001) trazem uma análise sobre a eficiência no uso de agentes de software em leilões do tipo inglês (ganha, a mais alta oferta). Os autores observaram que vários produtos semelhantes são leiloados em diversos sites diferentes e, até mesmo, em diferentes “salas” dos mesmos sites. Acreditaram que o fato dos agentes ficarem presos a um só leilão diminui a eficiência da compra. Para provar a teoria, criou-se um sistema em que o agente participa em vários leilões ao mesmo tempo. Como esperado, houve um aumento de eficiência para o comprador e, aparentemente paradoxal, também para o mercado. Contudo, ficou claro que à medida que aumentam os compradores munidos de tais mecanismos, só os vendedores que aumentam sua eficiência.

Há uma limitação para o sistema relatado no artigo: os leilões têm que ser concorrentes, ou seja, acontecem no mesmo tempo. Para leilões que começam em diferentes tempos, necessita-se de um mecanismo onde se possa avaliar se uma oferta atual é melhor que uma possível futura oferta em outro leilão. O caminho apontado para solucionar tal questão é um mecanismo de aprendizagem que utiliza a teoria da utilidade esperada para avaliar a possibilidade de um leilão mais recente ser mais eficiente que um leilão mais antigo ou vice-versa.

2.4 SISTEMAS MULTIAGENTES PARA O COMÉRCIO

Guttman e Maes (1998) descrevem alguns sistemas multiagentes de negociação de compras. Dividem-nos em dois tipos: os que negociam competitivamente e aqueles que negociam cooperativamente. Na negociação cooperativa, citam duas formas de praticá-la: por funções multiatributos de utilidade e pelo método, em inglês, *Distributed Constraint Satisfaction*, que analisa a decisão qualitativamente.

Muitas pesquisas concentram-se na construção de protocolos de negociação entre agentes para o comércio. O sistema Kasbah (CHAVEZ e MAES, 1996) implementa um protocolo simples para um sistema multiagente que tem três tipos de componentes: agentes de compra, de venda e o *Marketplace*. Este último é o mediador e controlador de todo o sistema. É o *Marketplace* que combina compradores e vendedores que desejam comercializar os mesmos produtos. A fase de negociação, entretanto, é feita diretamente pelos agentes de

compra e venda. Só depois de terminado o processo, com ou sem sucesso, que os agentes retornam a comunicação com o *Marketplace*, para registrar os acontecimentos.

Essa arquitetura do sistema, com agentes compradores, vendedores e um agente mercado, também foi utilizada na modelagem deste trabalho, que se diferencia por ter um agente mercado pertencente ao comprador (não neutro), com diversas outras responsabilidades. Isso na prática é possível quando o comprador é bastante grande, de forma que pode impor sua estrutura para os possíveis fornecedores.

Uma extensão do modelo *Kasbah* pode ser visto em De Paula (2000), que apresenta um modelo original de negociação bilateral, introduzindo várias facilidades de uma negociação real, como a sugestão de produtos alternativos, geração de ultimatoss e acordos locais em negociação de múltiplos atributos. As principais discussões deste trabalho foram: como modelar propostas, como avaliá-las, quais ações estão disponíveis para um negociador autônomo e qual ele deve tomar a cada rodada. A modelagem do sistema foi feita através da Metodologia Gaia (WOOLDRIDGE, JENNINGS E KINNY, 1999; WOOLDRIDGE, JENNINGS E KINNY, 2000). A dúvida que ficou na leitura do trabalho foi se a existência anunciada e prevista de ultimato pode levar a uma estratégia diferente de negociação que, possivelmente, venha prejudicar todo o processo.

Outros protocolos de negociação foram sugeridos, como, por exemplo, em Soo (2000), que também estabelece uma terceira parte na negociação.

Faraco (1998) sugeriu uma arquitetura de agentes para negociação, visando a utilização no comércio eletrônico. Consiste, assim como o *Kasbah*, de três tipos de agentes (clientes, fornecedores e roteadores) e suas inter-relações. Os atributos dos produtos em negociação são preço e quantidade. Os agentes fornecedores trabalham diretamente com as bases de dados dos fornecedores e utilizam regras preestabelecidas de negociação.

Uma outra arquitetura de um sistema multiagente para compra e venda é a relatada em Giese (1998). Tem-se um sistema que busca contemplar todo o processo de compra (em uma visão entre consumidores finais e varejistas), desde a pesquisa de mercado, para o comprador (consumidor), ao acompanhamento pós-venda, para o vendedor. Há um agente que avalia e ordena as alternativas de compra, através de tomada de decisão difusa. A decisão de comprar ou não, todavia, não é feita pelo agente. O sistema apóia a decisão, não a toma.

Para o processo de licitação, Kern (1998) propôs uma estrutura multiagente, buscando automatizá-lo. Mais um trabalho que mostra a potencialidade da tecnologia de agentes. A escolha da melhor alternativa é baseada no preço. Das propostas que atendem todos os pré-requisitos, a, com o menor preço, será a escolhida, o que é o processo comum em licitações.

Mais um trabalho que traz uma proposta de sistemas multiagentes para compras e vendas em Karacapilidis e Moraitis (2001). Uma das principais características desse trabalho é ter idealizado o agente comprador que “vive” mesmo após o término de uma compra. Na visão dos autores, criam-se empregados artificiais. Estes agentes têm também a capacidade de, proativamente, sugerir produtos para seus “empregadores”.

Os autores utilizam alguns recursos para modelar o sistema, como mostrar as arquiteturas dos agentes, ou utilizar os diagramas de Interação e de Atividades da UML. Mais precisamente, usou-se uma extensão da UML para uma boa representação. Os mecanismos de decisão são um pouco diferentes entre os agentes compradores e vendedores. O algoritmo de decisão é baseado nos pesos de cada um dos atributos de compra, dados pelos usuários responsáveis.

Há também, na literatura, sistema multiagente para compra e venda que foi pensado para toda a cadeia de suprimentos (CHEN et al., 1999). Cada vez que há necessidade de se comprar um suprimento, um agente é delegado para fazer tal compra. Essa necessidade percorre toda a cadeia, criando, também, uma cadeia de agentes. Assim como um agente comprador escolhe o melhor agente fornecedor, um agente fornecedor também tem a capacidade de escolher com quais agentes compradores negociar. Há uma boa discussão sobre linguagens de agentes e como representar o processo de negociação entre os agentes através de Redes de Petri Coloridas.

2.5 CUIDADOS PARA O COMÉRCIO ELETRÔNICO

Sabe-se que quando alguma empresa resolve vender produtos pela Internet, novos desafios ela terá que tomar. Um grande problema é a concorrência, que, na Internet, torna-se ainda mais perigosa, devido à facilidade do consumidor “ir” de um lugar para outro – está tudo à distância de um “clique”.

Um poderoso aliado para pesquisar os melhores produtos e os melhores preços para os consumidores é o agente de software, como vem sendo discutido neste texto. Não só pela capacidade de busca e sugestão de produtos, mas, em especial, pela capacidade de decisão.

Em Conway e Koehler (2000), há uma excelente discussão sobre as possibilidades advindas do uso de tal tecnologia. Inicialmente, comentam o uso de agentes que buscam, em vários sites, um determinado produto, classificando, por preço, a lista dos sites. Ao perceberem os problemas causados por tais tipos de mecanismos, os desenvolvedores dos sites resolveram dificultar seu uso. A estratégia utilizada foi tornar gráficas todas as informações sobre busca no site e sobre os produtos. Assim, esperava-se que só os usuários humanos pudessem entender, uma vez que os primeiros usuários “agentes de software” faziam inferências textuais pelo próprio código-fonte das páginas (HTML, por exemplo). Para a infelicidade dos desenvolvedores dos sites, começaram a surgir agentes que interpretam gráficos, através de mecanismos de Inteligência Artificial, bem como simulavam o uso do teclado e do mouse, de forma que não se perceba que não é um ser humano que está usando o site.

Como se não fosse suficiente o problema com a concorrência, os autores despertam para um problema ainda maior, associado ao uso de agentes de software no comércio: melhores decisões. Muitas vezes, os seres humanos são incapazes de garantir um bom mecanismo de decisão. Com a implementação de mecanismos automatizados de decisão em agentes, os consumidores podem atingir maior qualidade em suas escolhas, como já discutido em alguns casos para leilões.

O artigo traz um exemplo do uso de agentes, munidos de mecanismos de decisão, que mostram o potencial desta associação. Foi desenvolvido um agente de software para jogar “*blackjack*” em cassinos *on-line*. O “*blackjack*” é um jogo de cartas que tem um valor esperado positivo, quando se joga de forma ótima. Isto significa dizer que se um jogador faz jogadas ótimas (as melhores possíveis com as informações disponíveis), ao passar do tempo, ele ganha mais do que perde. Jogadores muito experientes apenas se aproximam de jogar otimamente. Assim, os cassinos ganham, em média, mais do que perdem com os apostadores, mesmo com a existência de alguns jogadores experientes. Entretanto, para um software, não é difícil jogar de forma ótima o “*blackjack*”. Desta forma, construiu-se um agente de software

que tem algoritmos para jogadas ótimas no “*blackjack*”. Resultado, os autores do artigo provaram que seu agente levaria o cassino a bancarrota.

O que se pode aprender em toda essa discussão é que fazer comércio por meios eletrônicos implica cuidados novos, pois não só as distâncias desaparecem, como se abrem as portas para mecanismos mais precisos de decisão e negociação.

2.6 MODELANDO AGENTES

Os progressos da Engenharia de Software, nas últimas duas décadas, foram principalmente feitos pelo desenvolvimento de abstrações mais poderosas e naturais para modelar e implementar sistemas complexos (WOOLDRIDGE, JENNINGS E KINNY, 1999).

Para modelar sistemas baseados em agentes, já há diversas técnicas propostas na literatura. Nesta seção, serão apresentadas algumas delas. Para alguns autores, agentes são abstrações completamente diferentes às existentes. Assim, propuseram técnicas completamente novas (WOOLDRIDGE, JENNINGS E KINNY, 1999; WOOLDRIDGE, JENNINGS E KINNY, 2000; JO, 2001; BRESCIANI et al., 2001). Outros autores acreditam que agentes são muito parecidos com abstrações já existentes, em particular, com objetos. Assim, propuseram extensões de técnicas baseada em objetos (KARACAPILIDIS e MORAÏTIS, 2001; BAUER, MÜLLER e ODELL, 2001; ODELL, PARUNAK e BAUER, 2001).

Duas técnicas de modelagem de sistemas multiagentes serão detalhadas aqui. A Gaia, que pertence a primeira corrente – que defende uma técnica completamente nova – e a *Agent Unified Modeling Language* (AUML), que reúne propostas de extensão para UML, uma já consagrada linguagem de modelagem de sistemas baseados em objetos, representando a segunda corrente.

2.6.1 A Metodologia Gaia

A Gaia foi proposta por Wooldrige, Jennings e Kinny (1999) como uma metodologia para desenvolvimento específico de sistemas baseados em agentes. Toda esta seção foi adaptada deste último artigo mencionado.

A metodologia Gaia compreende as fases de análise e projeto. Não há menção alguma sobre as fases seguintes da construção de um software (implementação, testes, implantação e manutenção).

Sistema, no jargão Gaia, é entendido como uma sociedade ou organização de agentes. Assim, quando se modelam sistemas, está se modelando uma sociedade de agentes. Sistema é o nível mais alto da hierarquia dos conceitos da análise. O próximo nível é o de papéis, que representa um conjunto de papéis que agentes da sociedade podem assumir. No mundo dos agentes, cada um desempenha um ou mais papéis, de forma semelhante a funcionários de uma empresa que podem assumir papéis de presidente, secretária, analista, etc.

Um papel é definido por três atributos: responsabilidades, permissões e protocolos. Responsabilidades determinam as funções que cada papel tem. Por exemplo, um papel “presidente” poderia ter as responsabilidades de divulgar resultados financeiros para os acionistas, de tomar decisões sobre aumentar ou diminuir a produção, etc.

Há dois tipos de responsabilidades: propriedades vitais e propriedades de segurança. As primeiras descrevem as ações e os estados que os agentes devem buscar, de acordo com as condições do ambiente. Já as segundas, indicam os estados que os agentes sempre têm que manter.

Suponha-se um agente comprador. Uma propriedade vital pode ser buscar fornecedores para um determinado produto. Já uma propriedade de segurança pode ser o de não comprar este produto por um valor maior que o especificado pelo criador do agente.

As permissões são os direitos que os papéis têm sobre determinados recursos. Já os protocolos são as formas como um determinado papel relaciona-se com outros.

Os conceitos apresentados acima são denominados Conceitos da Análise e seguem a seguinte hierarquia (Figura 2):

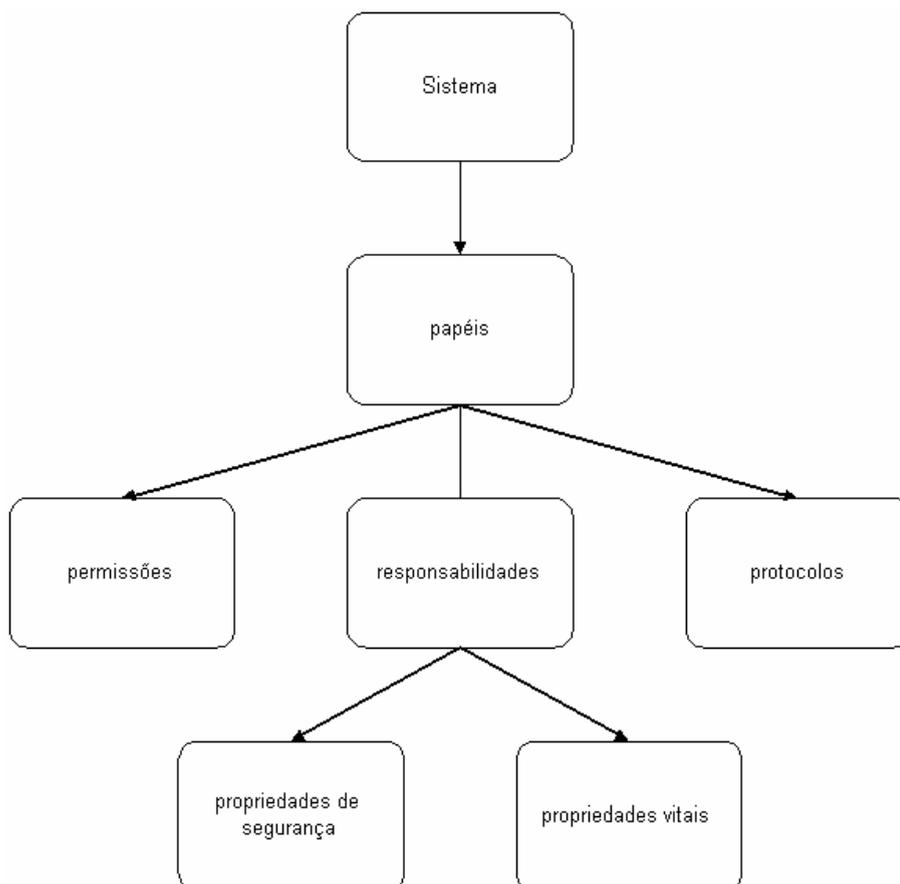


Figura 2 - Conceitos de Análise da GAIA

A fase de análise compreende a modelagem dos papéis e a modelagem das interações entre estes papéis. O processo de análise pode ser resumido da seguinte forma:

- a) identificar os papéis do sistema (saída: protótipo do modelo de papéis);
- b) para cada papel, identificar e documentar os protocolos associados, que são padrões de interação que ocorrem no sistema entre vários papéis (saída: modelo de interações);
- c) usando o modelo de interações, elaborar o modelo definitivo de papéis;
- d) repetir os passos “a”, “b” e “c”, até chegar a um nível satisfatório de análise.

A fase de projeto é, tradicionalmente, responsável pela transformação dos modelos abstratos da análise em modelos suficientemente de baixo nível para serem facilmente codificados. Na Gaia, entretanto, a fase de projeto tem como objetivo transformar os modelos abstratos da análise em modelos suficientemente fáceis de serem aplicados em outras

metodologias de desenvolvimento de software, como as orientadas a objetos. Isto se deve ao fato de os autores não trabalharem com linguagens de programação específicas para agentes.

A fase de projeto é constituída pela construção de três modelos: de agentes, de serviços e conhecimento.

O Modelo de Agentes tem como propósito documentar os vários tipos de agentes que serão usados pelo sistema em construção e, as instâncias de agentes que surgirão em tempo de execução. Para cada tipo de agente, definem-se os papéis que o constituem. Há um diagrama próprio para visualizar os papéis que cada agente tem.

O Modelo de Serviços identifica as funções desempenhadas para cada papel de agente, especificando suas principais propriedades.

O Modelo de Conhecimento simplesmente define as ligações de comunicação existentes entre os diversos tipos de agentes.

A Figura 3 mostra os relacionamentos entre os modelos da Metodologia Gaia.

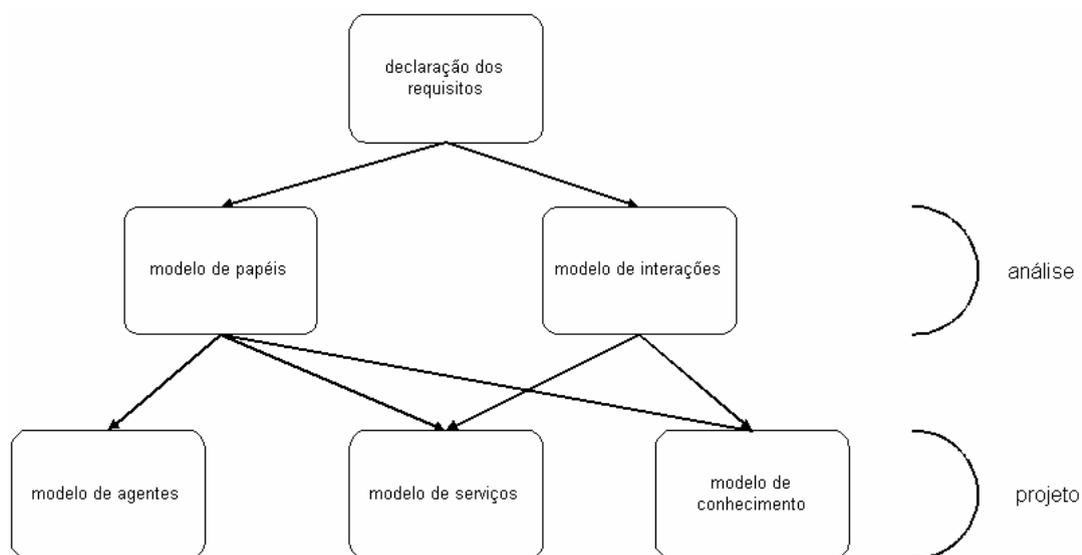


Figura 3 - Relacionamento entre os modelos da GAIA

2.6.2 Agent UML

Há uma organização (<http://www.auml.org>) que reúne pessoas que desejam estender a *Unified Modeling Language* (UML) para adequá-la aos conceitos e necessidades inerentes a

agentes – A *Agent UML* (AUML). No site desta organização, encontram-se diversos artigos sobre o tema. Em geral, cada artigo trata de algum diagrama em especial da UML, sugerindo algumas extensões para um bom uso com agentes de software.

É importante lembrar que a UML é uma linguagem para especificação, documentação, visualização e desenvolvimento de sistemas orientados a objetos. Surgiu da união de alguns dos maiores autores e suas metodologias sobre análise e projeto de sistemas baseados em objetos. Hoje é, sem dúvida, a linguagem de modelagem mais difundida para desenvolvimento de software, tanto no meio acadêmico, como nas empresas. Muito de seu sucesso é devido às ferramentas que apóiam os desenvolvedores durante as fases de construção de um programa. Quase todas “falam” UML. Por causa de sua popularidade, muitos autores acreditam que estendê-la para agentes é melhor que criar uma linguagem completamente nova.

Para melhor modelar os diversos aspectos dos sistemas – sejam estáticos, dinâmicos, de implementação – a UML provê vários diagramas: Diagramas de Casos de Uso, Diagramas de Classes, Diagramas de Objetos, Diagramas de Interação (Seqüência e Colaboração), Diagrama de Gráficos de Estados, Diagramas de Atividades, Diagrama de Componentes e Diagramas de Implantação. Além disso, a UML provê alguns mecanismos de extensão, como estereótipos e restrições. Maiores detalhes sobre a UML podem ser vistos na especificação da linguagem (UML 1.4 Specification, 2002).

Bauer, Müller e Odell (2001) preocuparam-se com os diagramas de interação da UML: Seqüência e Colaboração. Eles acreditam que os mecanismos de interação são o coração da *Agent UML*. Dentre as extensões sugeridas, encontram-se a inserção do conceito de papéis, com a abordagem de agentes, estruturas para declarar *threads* de interação, protocolos intervalados e aninhados, e nova semântica para as setas que representam as interações. Os autores relatam a experiência de desenvolvimento de um sistema baseado em agentes, onde houve uma boa aceitação das extensões da UML. Um artigo mais detalhado, mas com o mesmo teor, é encontrado em Odell, Parunak e Bauer (2001).

De encontro aos esforços dos autores acima mencionados, Lind (2001) demonstra como representar, com a UML padrão, protocolos de interação entre agentes, utilizando apenas os próprios mecanismos de extensão oferecidos por ela. Há uma preocupação em não

estender a UML, a fim de evitar dialetos da mesma. Contudo, Lind admite que a UML não é perfeita para a modelagem de agentes.

Buer (2001) propôs extensões para os Diagramas de Classe da UML, de forma a adequá-la à realidade dos sistemas baseados em agentes. As mudanças são tantas que se criou um novo diagrama, chamado Diagrama de Classes de Agentes, onde são inseridos os conceitos de papéis, ações, capacidades, protocolos, etc.

Outras sugestões de adaptação do Diagrama de Classes da UML também foram feitas, como a criação do Diagrama de Ontologia, que declara formalmente como será feita a comunicação entre os agentes (BERGENTI e POGGI, 2000).

Outras informações sobre a AUML podem ser vista diretamente no site da organização (www.auml.org), bem como na modelagem do sistema multiagente proposta nesta dissertação, no Capítulo 5.

3 COMPRAS, DECISÃO E NEGOCIAÇÃO

Neste capítulo, serão discutidos conceitos pertinentes à construção do modelo do sistema proposto nesta dissertação. A seção COMPRAS traz motivações para um processo de compras e suprimentos mais eficiente. O levantamento de quais são as variáveis importantes para um processo decisório de compras também é apresentado. Seguindo o raciocínio, o próximo assunto é DECISÃO, que pode ser tratada sob vários enfoques. O enfoque deste trabalho é apresentado de forma que o mecanismo de decisão do modelo proposto seja mais bem entendido. A última seção, NEGOCIAÇÃO, apresentará a abordagem de como será tratado este processo no corrente trabalho, facilitando também a compreensão do algoritmo de decisão proposto.

3.1 COMPRAS

3.1.1 A importância da Função Compras para a organização

Por motivos internos e externos à organização, como vantagem competitiva, avanço tecnológico, políticas governamentais e de blocos econômicos, recursos finitos, maior proporção de gastos fora da empresa, concentração das compras em poucos e grandes fornecedores, aumento da consciência ambiental, a função compras vem aumentando sua importância e reconhecimento. O reconhecimento de que o processo mais eficiente de compras e suprimentos pode aumentar o faturamento das empresas também contribui para uma maior atenção à função (BAILY et al., 2000).

3.1.2 A Função Compras na Administração da Produção

Na Figura 4, adaptada de Slack et al. (1997, p.414), pode-se ver toda a Função Compras – na ótica da Administração da Produção – desde as requisições, ao recebimento dos produtos e serviços. As primeiras etapas da função são de particular interesse para os propósitos deste trabalho, pois são nelas que se pode aplicar o sistema modelado em capítulo posterior. Estas etapas estão em destaque na Figura 4, delimitadas por uma linha tracejada.

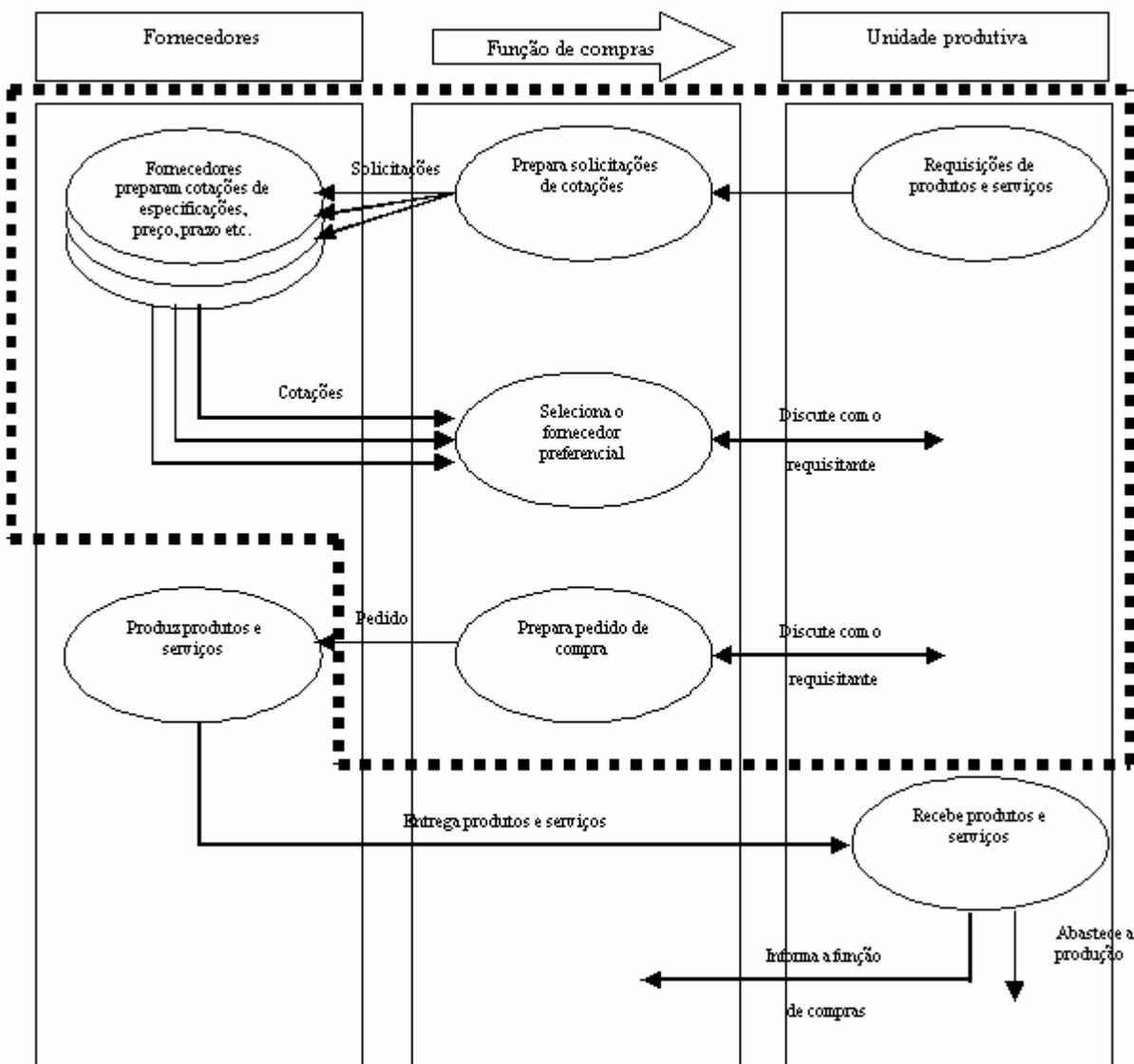


Figura 4 - Função Compras (SLACK et al., 1997)

Foram modeladas todas as etapas da “Requisições de produtos e serviços” à “Prepara pedido de compras”, sendo “Seleciona o fornecedor preferencial” e “Discute com o requisitante” (negociação), além de modeladas, implementadas.

3.1.3 Tipos de itens de compra

Há uma grande variedade de itens de compra possíveis, desde simples produtos de escritório a complexos itens e serviços (GEBAUER e SEGEV, 2000). Uma boa classificação

do processo de compras (adaptada de GEBAUER, BEAM e SEGEV, 1998), de acordo com o tipo de itens relacionados, é:

- a) *compras de matérias-primas e bens de produção*. Usualmente caracterizadas por grandes quantidades, alta frequência e importantes e únicas especificações. Entrega *just-in-time* é frequentemente crítica;
- b) *compras de itens de manutenção*, de reparo e operacionais, MRO (do inglês: *Maintenance, Repair and Operating*). São usualmente caracterizadas pelos baixos custos unitários e baixos volumes, mas relativamente com alta frequência. Exemplificando, ter-se-ia suprimentos para escritório;
- c) *compras de bens de capital e compras esporádicas*. Significa lidar com bens de alto valor em baixa frequência (exemplo, nova fábrica) e/ou comprar itens fora do processo regular de compras, geralmente por causa de conveniência ou velocidade de aquisições.

3.1.4 Variáveis-chave de compras

Para se poder escolher a melhor opção de compras dentre possivelmente muitas, necessita-se de variáveis de decisão. Com um mecanismo de decisão apoiado nos valores destas variáveis, é possível estabelecer uma ordem de preferência dentre as diversas cotações de compra. O processo de negociação também é apoiado na manipulação de tais variáveis.

Baily et al. (1998) estabelece as seguintes variáveis-chave de compras:

- a) *qualidade*, que pode ser definida como “adequação ao propósito”;
- b) *quantidade correta*, onde se levam em consideração, principalmente, os custos de estoque e custo de falta do produto. A quantidade correta é aquela que minimiza os custos, de uma forma global;
- c) *tempo*. Está relacionado com os custos envolvidos no planejamento, atraso e necessidade de alternativas contingenciais;
- d) *fonte de suprimento*. Deseja-se verificar quais os fornecedores mais adequados, levando em consideração os bons atributos de um fornecedor, bem como a análise de fatores

estratégicos, por exemplo, a quantidade de fornecedores para determinado produto, possíveis parcerias ou preferências;

e) *preço*. Há uma inferência sobre quais os preços razoáveis para cada um dos produtos, mas, isoladamente pensando, deseja-se o menor preço. É importante ressaltar que a decisão de compra é tomada a partir da análise de todas essas variáveis. Os preços podem mudar de acordo com as outras variáveis, por exemplo;

f) *negociação*.

Outros autores sugerem diferentes enumerações das variáveis-chave. Em Degraeve, Labro e Roodhooft (2000), vê-se que muitas “dimensões” são importantes na escolha de um vendedor, incluindo *net price* (o preço global, somado o custo da transação), qualidade, entrega, performance histórica, capacidade, sistemas de comunicação, serviço, localização geográfica, etc.

Em Slack et al. (1997, p.413), volta-se a encontrar uma enumeração parecida para as variáveis-chave de compra (exceção: ausência da variável “negociação”), quando os autores estabelecem os objetivos da Função Compras, também denominados “os cinco corretos de compras”:

- a) ao preço correto;
- b) para entrega no momento correto;
- c) produtos e serviços da qualidade correta;
- d) na quantidade correta;
- e) da fonte correta.

3.2 ANÁLISE DE DECISÃO

3.2.1 Um panorama do processo de análise de decisão

Muitas vezes, quando se tem que decidir, defronta-se com problemas grandes e complexos, onde o processo intuitivo torna-se árduo, frequentemente pela incapacidade de entendê-los como um todo. Uma forma de buscar uma melhor compreensão do problema é

estruturá-lo em problemas menores, de forma a tornar a decisão inicial um conjunto de decisões menores e mais simples. A decisão intuitiva desses elementos é mais confiável.

A Figura 5 traz o processo de análise de decisão proposto por Clemen (1996):

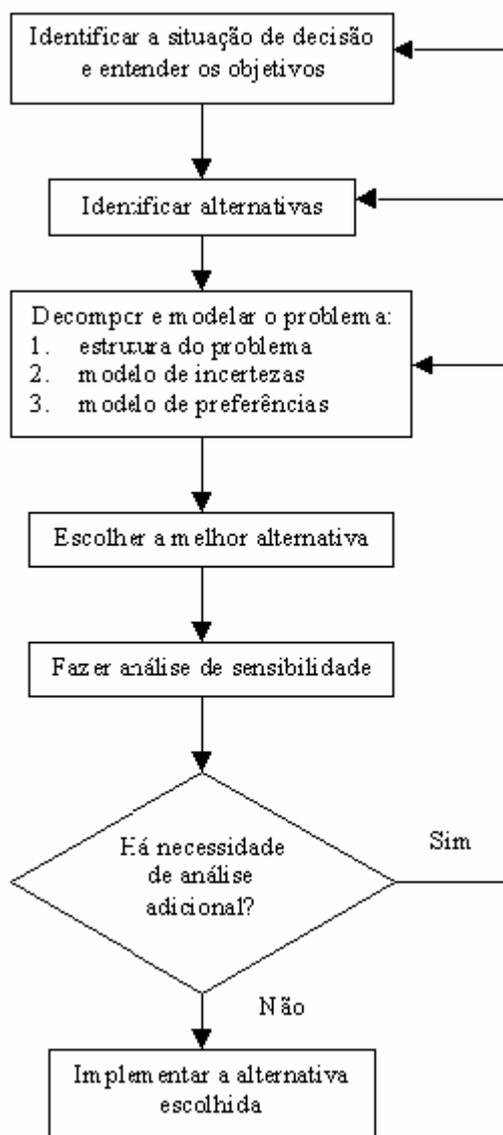


Figura 5 - Processo Decisório (CLEMEN, 1996)

Ainda segundo Clemen (1996, p.5-8), o primeiro passo para o tomador de decisão é identificar a situação de decisão e entender seus objetivos neste contexto. Na próxima etapa, o conhecimento dos objetivos pode ajudar na identificação das alternativas. Geralmente, um cuidadoso exame e análise dos objetivos podem revelar alternativas que não eram óbvias *a priori*. Os próximos dois passos podem ser chamados de modelando e solucionando. A idéia

de modelagem é crítica para o processo, assim como é em quase todas as abordagens quantitativas ou analíticas. De posse dos modelos de decisão, a escolha da alternativa deve ser facilitada. A análise de sensibilidade verifica se, mudando um pouco os aspectos do modelo, a decisão muda. Se afirmativo, pode-se desejar voltar para alguns dos passos anteriores, como indicado na Figura 5, caracterizando um processo iterativo. Por fim, implementa-se a alternativa escolhida.

3.2.2 Paradigma de Análise de Decisão segundo Keeney e Raiffa (1976)

Keeney e Raiffa (1976) resumiram um paradigma de análise de decisão em um processo de cinco passos: pré-análise, análise estrutural, análise das incertezas, análise de utilidade ou de valor e análise de otimização. Tal processo foi pensado de forma mais restritiva que o da seção anterior (CLEMEN, 1996), pois tinha o propósito explícito do uso de funções de valor ou de utilidade na análise de decisão. Apesar de mais restritiva, a revisão deste paradigma é pertinente por estar intrinsecamente ligado a funções de valor, que são fundamentais nos mecanismos de decisão e negociação sugeridos em capítulo posterior.

No passo intitulado pré-análise, identifica-se o problema e as alternativas viáveis são discriminadas. Analogamente pensando, corresponde aos dois primeiros passos do processo de análise de decisão de Clemen (1996).

Na *análise estrutural*, o tomador de decisão estrutura a anatomia qualitativa de seu problema. Quais escolhas podem ser tomadas? Quais não podem? Quais as seqüências de decisões? Quais informações podem ser apreendidas no decorrer do caminho? Quais as chances associadas a cada possível alternativa? Uma ferramenta bastante usada para estruturar o problema é a árvore de decisão, que são grafos compostos por arcos e dois tipos de nós. Os nós podem ser de decisão ou de chance. Os arcos indicam as possíveis alternativas ou conseqüências.

A *análise de incertezas* é o passo em que o tomador de decisão atribui probabilidades aos arcos que partem dos nós de chance. Estas probabilidades podem ser feitas através de dados estatísticos históricos, inferências através de modelos estocásticos ou dinâmicos, através da opinião de especialistas ou por julgamentos pessoais. As probabilidades atribuídas devem ser verificadas em relação a sua consistência interna.

Na *análise de valor ou de utilidade* o tomador de decisão atribui valores de utilidade para conseqüências e loterias associadas aos caminhos da árvore de decisão. Em uma análise real, para cada caminho, há associados vários custos e benefícios econômicos e psicológicos.

Na *análise de otimização*, tendo-se em mãos todos os valores de utilidade associados a cada uma das alternativas e/ou loterias, o tomador de decisão calculará todos valores de utilidades globais, ou seja, por todos os caminhos possíveis da árvore de decisão. Desta forma, poderá escolher a melhor estratégia – aquela que maximize a utilidade esperada.

3.2.3 Teoria da Utilidade Esperada

A Teoria da Utilidade Esperada tem suas raízes no século XVIII, com estudos feitos por B. Pascal e D. Bernoulli. Foi motivada pela procura do entendimento dos problemas de decisão envolvidos em jogos de azar.

Somente na primeira metade do século XX a Teoria da Utilidade Esperada foi resgatada. Desta vez, a teoria veio formulada por um conjunto de axiomas e teve uma forte aceitação na comunidade acadêmica da época, sendo tema presente até nossos dias. Devem-se as pedras fundamentais da teoria aos trabalhos de Ramsey (apud BECKER, 1988) e von Neumann e Morgenstern (1944).

3.2.4 Axiomatização

Não há somente uma axiomatização da teoria vigente. Becker (1988) discute a respeito, elegendo, em seu texto, o conjunto de axiomas de Herstein e Milnor (apud BECKER, 1988) para segui-lo de forma próxima, por considerar a solução mais elegante do ponto de vista matemático.

Duarte (2001) sintetiza os axiomas básicos da moderna Teoria da Utilidade de von Neumann e Morgenstern (1944), como mostrado na seguinte adaptação:

a) Axioma da ordenação preferencial

O sujeito é capaz de comparar e ordenar pares de alternativas. Desta forma, existe uma função transitiva e reflexiva (\succ - preferido a, ou \succeq - preferido ou indiferente a) que permite ordenar alternativas.

b) Axioma da continuidade

Para três alternativas diferentes p , q e r , se r é preferível a q e q é preferível a p ($r \succ q \succ p$) existe um real I para qual $I r + (1 - I)p \approx q$, onde \approx é a operação de indiferença.

c) Axioma da independência preferencial

Para três alternativas diferentes p , q e r e um real I , se $p \succ q$ então toda combinação convexa $I p + (1 - I)r$ é preferível a toda combinação convexa $I q + (1 - I)r$.

Com estes axiomas é possível determinar que se alguém prefere uma alternativa r a s , então o valor retornado pela função de utilidade de r é maior que o valor retornado pela mesma função aplicada a s . Matematicamente, tem-se: $u(r) > u(s)$.

3.2.5 Diferença entre função de valor e função de utilidade

Ambas especificam uma estrutura única de preferência. A função de valor, entretanto, é um caso particular de função de utilidade. A particularidade é que a primeira está inserida em um ambiente de certeza, enquanto a segunda também pode tratar problemas que envolvam loterias.

Em Keeney e Raiffa (1976, p.220) há uma breve discussão a respeito da nomenclatura utilizada para diferenciar as duas: função de valor e de utilidade. Eles utilizam a nomenclatura como aqui citada, mas advertem que a função de valor também é rotulada como função de utilidade ordinal, função preferencial, *worth functions*, função de utilidade de Marshallian e, até mesmo, de função de utilidade. Enquanto a função de utilidade é chamada também função de utilidade probabilística, função de utilidade cardinal, função preferencial, função de utilidade de von Neumann e, também, apenas, função de utilidade.

3.2.6 Alternativas com múltiplos atributos

Algumas vezes, há diversas alternativas para se avaliar com apenas um atributo. Poder-se-ia imaginar uma situação de compra de um determinado aparelho de TV. Uma vez que já se saiba exatamente qual o modelo desejado, pode-se decidir com apenas um atributo: preço, por exemplo. Contudo, situações semelhantes, onde se tem apenas um atributo, não são comuns. O mais fácil de se encontrar são problemas que exigem a avaliação de mais de um

atributo. Por exemplo, as variáveis-chave de decisão de compra, anteriormente mencionadas, podem formar, cada uma delas, um atributo. Para estabelecer uma ordem de preferência dentre possíveis muitas alternativas, ter-se-ia que utilizar funções de utilidade multiatributos (caso deseje-se utilizar funções de utilidade como ferramenta de decisão).

Seja A uma alternativa constituída por um conjunto de atributos $\{X_1, X_2, \dots, X_n\}$, onde n é o número de atributos. Uma função para representar a utilidade u de tal alternativa teria a forma:

$$u(x_1, x_2, \dots, x_n) = f[u_1(x_1), u_2(x_2), \dots, u_n(x_n)]$$

Onde x_i é uma quantidade específica de X_i , f é uma função escalar e u_i é uma função de utilidade sobre X_i .

Dependendo da natureza do problema, envolvendo incerteza ou não, e das relações de dependência entre os atributos, a função de utilidade assumirá algumas diferentes formas. A forma (talvez, formas) que será incorporada nos agentes será determinada no decorrer da pesquisa.

3.2.7 Dominância

Uma alternativa A_1 domina uma alternativa A_2 se todos os níveis de seus atributos são maiores ou iguais aos níveis dos atributos de A_2 , com, ao menos, um deles estritamente maior.

3.2.8 Fronteira eficiente

Fronteira eficiente é o subconjunto que não é dominado do conjunto de conseqüências possíveis para todas as alternativas. Se o conjunto de conseqüências de uma alternativa está contido na fronteira eficiente, não há melhor alternativa para o problema, no máximo há uma indiferente.

3.2.9 Curvas de indiferença

Curvas de indiferença podem ser pensadas como um conjunto de alternativas (algumas talvez sendo hipotéticas) entre as quais o tomador de decisão é indiferente (CLEMEN, 1996, p.540).

3.2.10 Paradoxos

A análise de decisão, incluindo o uso de funções de utilidade, objetiva facilitar a tomada de decisão, estruturando-na em decisões menores e mais fáceis de lidar. Contudo, mesmo decisões aparentemente simples podem levar a resultados confusos e, algumas vezes, o tomador de decisão entra em contradição com o que havia expressado anteriormente.

Pesquisas da psicologia, principalmente, demonstram que a suposta “racionalidade” do processo de decisão apoiado na Teoria da Utilidade Esperada é, em alguns casos, paradoxal. Mudando-se o cenário, a decisão também pode mudar. Por exemplo, em problemas que envolvem vida. A frase “chance de sobrevivência de 80%” trocada por “chance de morte de 20%” pode fazer com que o tomador de decisão modifique sua ordem de preferência. Clemen (1995, p.510) exemplifica alguns paradoxos conhecidos e discute sobre as implicações dos mesmos.

3.3 NEGOCIAÇÃO

Negociação vem sendo definida como uma forma de tomada de decisão envolvendo duas ou mais partes que não podem tomar decisões independentemente e são requisitadas a fazer concessões para chegarem a um acordo (KERSTEN et al. apud JAIN e SOLOMON, 200).

Existem várias perspectivas para a negociação, perspectivas psicológicas, sociais e matemáticas, por exemplo. Concentrar-se-á, aqui, na perspectiva matemática. A justificativa é que se deseja automatizar o processo de negociação de compras. Para automação, precisa-se de modelos bem definidos e de implementação viável em computadores.

3.3.1 Negociação em compras

Há uma extensa bibliografia a respeito da teoria e prática de negociação. Em Baily et al. (1998), por exemplo, há um capítulo do seu livro dedicado à negociação de compras. De início, os autores ressaltam que a negociação nem sempre é a maneira apropriada para atingir um acordo. Em alguns momentos, é possível usar a persuasão, a aceitação total, a coerção ou a solução do problema como alternativa.

Os autores estabelecem três fases no processo de negociação: preparatória, reunião e final, com especial atenção à primeira fase. Aspectos sobre a qualidade de um bom

negociador são evidenciados no texto, havendo até uma análise da linguagem corporal ideal em uma negociação.

3.3.2 Negociação integrativa

Raiffa (1982, p.131) define “*integrative bargain*” como sendo a barganha (negociação) em que duas partes e várias questões estão envolvidas.

Negociação integrativa também é chamada de negociação colaborativa, que vem sendo descrita como o processo de tomada de decisão para resolver um conflito envolvendo duas ou mais partes sobre objetivos múltiplos, mas não mutuamente exclusivos (LEWICKI, SAUNDERS e MINTON apud GUTTMAN e MAES, 1998). Ou ainda, na teoria dos jogos, é descrita como um jogo de soma não-zero, onde os valores ao longo das múltiplas dimensões podem ser deslocados em diferentes direções, possibilitando que todas as partes alcancem melhores resultados (ROSENSCHEIN e ZLOTKIN apud GUTTMAN e MAES, 1998).

Para cada questão envolvida na negociação, que se encarará como um atributo de decisão, existem alguns níveis que o negociador deve estipular. Um deles é o nível de reserva, que é aquele mínimo, do qual não se pode baixar, ou máximo, que não se pode elevar – no caso do valor de um produto em negociação. Em uma negociação de compra e venda, o vendedor deve estabelecer um valor mínimo pelo qual ainda aceita vender. Já o comprador tem o seu preço de reserva, ou seja, o valor máximo que está disposto a pagar por aquele determinado produto em negociação. Estes níveis dos atributos são valores restritivos.

Outros níveis importantes dos atributos são os níveis desejados. Ainda no exemplo de compras, determinado comprador pode desejar receber o produto em negociação no dia seguinte, como ideal, mesmo tendo um prazo de reserva de sete dias, por exemplo. Já o vendedor pode desejar entregar em quinze dias, mas tem como reserva cinco. Em menos que isto, não tem como entregar.

Continuando o exemplo do prazo, é interessante notar que há como se chegar a um acordo, pois há o que se chama de *zona de acordo* entre o quinto e o sétimo dia – sendo um prazo viável para ambos: comprador e vendedor.

A existência de vários atributos de decisão – como no caso de decisão de compras – favorece o alcance de um acordo entre duas partes envolvidas, já que fica mais fácil encontrar uma forma de ceder em algum atributo em troca de benefício em um outro.

Com o uso de funções multiatributos de utilidade, pode-se facilmente manipular o impacto de ceder em um atributo e ganhar em outro. A negociação pode fluir com maior rapidez e eficiência. É importante ressaltar, entretanto, que é difícil encontrar os dois negociadores munidos de tais funções.

Após os negociadores terem chegado a um acordo, podem tentar melhorar seus desempenhos juntos. Não se trata mais de um jogo de soma-zero, mas algo como “aumentar o bolo para depois repartir”.

No gráfico da Figura 6, tem-se traçadas as curvas de indiferença de dois negociadores. As curvas com os traços cheios são as do *negociador 1*, já as tracejadas são as do *negociador 2*. Para o *negociador 1*, as curvas mais a nordeste são melhores, enquanto para o *negociador 2*, ao sudoeste.

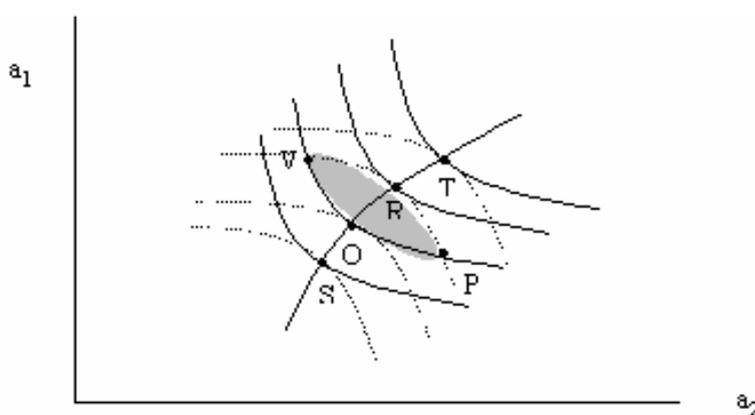


Figura 6 - Curvas de Indiferença e Curva de Contrato

Suponha-se o ponto P como representando o acordo que se chegou. Note-se que para o *negociador 1*, é indiferente ao ponto P qualquer um na curva PQV . De forma análoga, para o *negociador 2*, os pontos da curva PRV são indiferentes.

A área hachurada no gráfico indica todos os pontos possíveis que o acordo pode ser melhorado, ou tornar-se indiferente. A curva SQR pode ser entendida como a fronteira

eficiente da negociação, que nesta ambiência é rotulada de *curva de contrato*. Nesta curva, não há como melhorar a negociação de forma colaborativa (“repartir o bolo para repartir”), volta-se a um processo de soma-zero.

O desafio, portanto, é depois de se ter chegado a um acordo P qualquer, melhorá-lo, idealmente, para algum ponto da curva de contrato. Uma forma possível é através de um processo interativo e iterativo de *trade-offs* e concessões. Por exemplo, o *negociador 1* pode questionar o *negociador 2* se ele está interessado em ganhar um pouco do atributo a_1 em troca de ceder um pouco do atributo a_2 , que estão representados na Figura 6, no eixo das ordenadas e abscissas, respectivamente.

3.3.3 Computadores e negociação

Programas de computador vêm sendo usados para facilitar o processo de negociação. Tais programas foram denominados NSS, do inglês *Negotiation Support Systems*.

Dividindo-se os NSS em seus componentes básicos, teríamos um DSS e um EMS, do inglês *Decision Support Systems* e *Electronic Meeting Systems*, respectivamente. O primeiro é o componente de auxílio à decisão, já o segundo refere-se ao componente de comunicação. Um estudo comparativo entre negociação face a face e mediada por tais *softwares*, pode ser encontrado em Jain e Solomon (2000), discorrendo sobre a importância dos dois componentes, com especial atenção ao EMS.

O foco, entretanto, do presente trabalho não é a negociação auxiliada por computadores, mas um passo na direção de negociação feita sem a intervenção humana. Deseja-se que os responsáveis pelas compras e suprimentos estabeleçam critérios de negociação *a priori* e deleguem-na a programas de computadores. Desta forma, ter-se-á menor necessidade de tempo humano no processo, almejando menores custos para organização. Uma forma possível para tal feito é a utilização de *agentes de software*.

4 UMA OPORTUNIDADE PARA USO DE AGENTES

4.1 O CENÁRIO

A Brasil Telecom S.A. (BrT) é uma das maiores empresas de telecomunicações do país. Tem como área de atuação, hoje, os estados de Rio Grande do Sul, Santa Catarina, Paraná, Mato Grosso do Sul, Mato Grosso, Goiás, Acre, Rondônia e Tocantins, mais o Distrito Federal. Seu principal produto ainda é o telefone fixo, mas é crescente a participação dos outros produtos, como os de comunicação de dados e de Internet, nos resultados da empresa.

A empresa tem cerca de 10 milhões de linhas telefônicas instaladas, mais de 280 mil telefones públicos, contando com, aproximadamente, 7 mil funcionários – ou colaboradores – usando o jargão da própria empresa. Destes 7 mil, uns 700 estão na Matriz, em Brasília, onde as decisões estratégicas são tomadas. A maioria atua nas filiais, incluindo a filial do Distrito Federal, onde tratam de questões comerciais e operacionais.

A estrutura da Diretoria de Tecnologia da Informação ilustra bem o exposto acima. Só estão nas filiais as pessoas responsáveis pela operação e manutenção da infra-estrutura. Na Matriz, estão os colaboradores que tomam decisões sobre quais soluções serão implantadas, está a manutenção dos sistemas legados, o planejamento de TI, etc.

Das gerências da Diretoria de TI, uma tem especial destaque para este trabalho: a Gerência de Planejamento e Recursos de TI. O porquê disso é o fato que esta gerência inclui a área de Obtenção de TI da empresa, responsável por fazer a interface entre as diretorias de Materiais e Serviços e a de Tecnologia da Informação, em relação à aquisição de todos os bens, serviços e produtos relativos à Tecnologia de Informação.

Das funções realizadas pela área, cabe destacar as seguintes:

- a) *Requisição.* É a etapa operacional. Uma vez definido que recurso comprar e, sabendo-se da existência de saldo para realizar a compra, tem-se apenas que incluir os dados no

software de gestão (SAP). Ordens Internas são preparadas para lançamento no software. É de grande importância que estes dados estejam criteriosamente bem definidos. Também, nesta etapa, imobilizam-se as novas aquisições. Esta etapa é decorrente, entretanto, de muitas outras, onde o cenário não é simples. Fatores políticos, jurídicos, só para citar alguns, podem ser tão, ou mais, dispendiosos, tanto em relação a tempo, quanto a custos com pessoal.

- b) *Relacionamento*. Apesar de haver uma gerência para o relacionamento com os clientes internos da Diretoria, a equipe de Obtenção está sempre recebendo *e-mails* e telefonemas de pessoas que querem “agilizar” o processo. Este trabalho de relacionamento ocupa um bom tempo da equipe.
- c) *Contratos*. É também de responsabilidade da equipe de compras, a gestão de todos os contratos feitos com os fornecedores dos recursos de TI, incluindo a gestão de garantias, licenças, vigência e o fluxo financeiro.
- d) *Orçamento de TI*. A participação no levantamento das despesas e investimentos de TI é mais uma das responsabilidades da equipe. É feita anualmente, com acompanhamento mensal.
- e) *Fornecedor*. Todo o relacionamento (negociação de escopo, entrega, etc.) com os fornecedores, que diz respeito a pagamentos, recebimentos de bens, juros, multas, cálculos tributários, etc., é feito também pela equipe.

Há ainda outras funções desempenhadas, como a gestão do contrato de impressão das contas dos clientes da empresa e a gestão dos contratos de links com outras empresas de telecomunicações (links para uso exclusivo de projetos de TI).

Para quase todos os itens de compra de TI, a Obtenção é envolvida no processo de definição e negociação de compra, juntamente com a Diretoria de Materiais e Serviços. Dentre estes itens, estão desde simples suprimentos para reparos de computadores, como um teclado, a servidores de dez milhões de reais, por exemplo; de licença para *softwares shareware*, a aplicações caríssimas, como a solução de gestão empresarial da SAP.

Há certos itens de TI que podem ser classificados como de reparos, por exemplo, discos, teclados, *mouses*, cabos de par trançado, etc. Esses itens são solicitados pelos Núcleos

de TI para a Obtenção. São nesses núcleos que estão os técnicos responsáveis pela manutenção da infra-estrutura. Como exemplo, caso haja um microcomputador com a memória RAM defeituosa, o técnico do Núcleo solicita um pente de memória novo para reparar o problema, via *e-mail*.

Há uma diretriz na empresa para não haver estoque, ou o mínimo possível. Assim, necessita-se de rapidez na compra de tais itens. A solução encontrada para dar maior rapidez a estas compras, sem que seja necessária grande intervenção do pessoal de compras, seja da Obtenção de TI, seja da Diretoria de Materiais e Serviços (DMS), é o contrato aberto, também chamado de contrato guarda-chuva.

Um contrato aberto é quando se acorda com um fornecedor a compra de determinados produtos, em determinadas quantidades máximas, a um preço preestabelecido. Exemplificando, a Brasil Telecom fecha, com um fornecedor qualquer, um contrato aberto de compra de duzentos teclados de determinada especificação, por um preço unitário de R\$ 30,00. Isso não implica necessariamente a compra de duzentos teclados pela empresa, mas uma aprovação prévia da DMS de compra de até duzentos teclados (daquela especificação) ao fornecedor ganhador da concorrência, no máximo, ao preço preestabelecido, durante a vigência daquele contrato.

Para a empresa fornecedora, a vantagem é óbvia: o fornecimento de muitos produtos. Apesar de não ter a garantia que venderá todos aqueles produtos cotados, tampouco de que a empresa (no caso, a BrT) os comprará realmente dela, esse tipo de contrato é feito para não complicar. Assim, é “certo” que a empresa ficará comprando do fornecedor escolhido, ao menos para itens baratos, como os suprimentos que os Núcleos pedem. A empresa contratante (BrT) não assume responsabilidades, só a contratada. Há uma declaração de intenção de compras por parte da contratante, que, obviamente, já representa um grande negócio para a contratada.

A escolha da empresa fornecedora é feita através de uma concorrência, ganhando aquela que oferecer um valor total (somatório de todos os itens da lista, nas quantidades solicitadas) menor e, claro, obedeça a todos os critérios estabelecidos, como prazos de entrega, disponibilidade, etc. A busca pelos possíveis fornecedores e o contato com os interessados são feitos através de um Mercado Eletrônico (<http://www.me.com.br>), no caso da Brasil Telecom.

Este mercado eletrônico, no qual a Brasil Telecom participa, é um terceiro, que reúne compradores e fornecedores. O que o mercado faz, em essência, é fazer um cruzamento de quem procura determinado produto para comprar, com aqueles que vendem estes produtos. Depois de feito esse cruzamento, as empresas entram em contato entre si, não havendo qualquer decisão automatizada, mas apenas contatos por meios eletrônicos. Há, claro, inúmeras vantagens do uso de um mercado eletrônico, em relação ao mercado tradicional, como possibilidade de ampliação de parceiros comerciais, redução do fluxo de papéis, facilidades de formas de pagamento, agilidade na publicação de uma concorrência, diminuição de custos, etc.

Voltando aos contratos abertos, algumas questões surgem deste tipo de prática. Parece evidente que o processo de compra de suprimentos será facilitado e agilizado, diminuindo o tempo despendido pelo pessoal especializado em compras e dando a velocidade necessária para, neste exemplo, as equipes técnicas, que precisam reparar rapidamente os equipamentos.

Note-se também que a barganha neste procedimento é feita sobre o valor total de todos os itens, com as quantidades estimadas. Mas, como saber se as quantidades estimadas vão ser realmente as reais necessidades? Pode-se inferir que, ao final do período de vigência do contrato, as reais necessidades devem divergir das estimadas e a escolha do melhor fornecedor poderia ter sido outra.

Será que se a concorrência fosse feita produto por produto, o somatório de todos os itens seria menor? Até que ponto o fato de ser uma possível grande venda, o fornecedor diminuiria seus preços? O fato de o fornecedor comprometer-se a manter o preço durante toda a vigência do contrato não pode aumentar a cotação dos mesmos?

Diante de tantas questões, a mais evidente e suficiente para garantir a existência de contratos abertos é a agilidade. Há também o desejo de manter a compra de suprimentos de forma centralizada (como no caso da BrT). A centralização tem suas vantagens. Por exemplo, os núcleos não precisam entender de compras, tampouco perder tempo em decisão e negociação. Há também um maior controle da empresa quando se tem um processo centralizado.

Contextualizando para o caso das obtenções de suprimentos para os Núcleos de TI da BrT, tem-se: quando um técnico observa a necessidade de um item de reparo, faz a solicitação

deste item a equipe de Obtenção. Não importa em que Núcleo este técnico está alocado (há Núcleos em todas as filiais e, às vezes, mais de um por filial), o pedido é feito para a Obtenção de TI, que fica na Matriz. Caso o item esteja previsto no contrato aberto, tudo ocorrerá rapidamente, como necessário para o pessoal técnico, uma vez que a aprovação da compra foi tomada *a priori*.

Aprofundando a análise, a Obtenção de TI também não é a última palavra no processo de compras. Quem, de fato, negocia com o fornecedor e emite pedido é a Diretoria de Materiais e Serviços. Os contratos abertos também são criados e negociados pela DMS. A Obtenção participa definindo quais os itens necessários (especificações) e em que quantidade.

4.2 UMA SOLUÇÃO ALTERNATIVA PARA A COMPRA DE SUPRIMENTOS DE TI

A solução aqui apresentada ilustra um possível uso de agentes de software. As intenções principais são despertar, com um exemplo, para a utilização de tais tecnologias nas empresas, discutir o uso da UML estendida para agentes de software, como linguagem de modelagem, além de propor mecanismos para os processos de decisão e negociação, aplicados ao contexto da Compras.

Inicialmente, a equipe de Obtenção de TI estimaria as necessidades futuras de suprimentos de TI, elaborando uma tabela com as especificações dos produtos e as quantidades provavelmente necessárias. A DMS então, faria uma consulta, possivelmente através do mercado eletrônico, dos preços, incluindo-os na tabela dos produtos. Até aqui, o processo está idêntico ao atual.

Diferentemente do que ocorre hoje, o próximo passo não seria uma concorrência para escolher um fornecedor para todos os itens – um contrato aberto. A sugestão é substituir tal tipo de contrato pela delegação da decisão de compra e negociação da mesma a agentes de software. Todas as vezes que houvesse necessidade de compra de um suprimento, o Núcleo criaria um agente de software que solicitaria cotações aos fornecedores, ordenaria as alternativas e negociaria a compra com o vencedor da concorrência.

Fundamentalmente, estaria sendo feito um processo de descentralização de compras. Esta descentralização, entretanto, não implicaria necessidade de treinamento do pessoal dos Núcleos para compras, tampouco um descontrole da DMS ou da Obtenção sobre as compras na Brasil Telecom. O conhecimento do processo estaria embutido no agente. Já o controle das

compras estaria garantido pelas informações que os agentes reportariam. O treinamento do pessoal dos Núcleos seria pequeno, apenas mostrando como criar e configurar os agentes. Para quem trabalha com suporte técnico de informática, não seria um grande desafio.

Nesta abordagem, para cada produto, haveria uma concorrência, levando, possivelmente, a um custo total menor. Esta possibilidade, entretanto, não pode ser verificada, mas sim inferida, se o sistema multiagente fosse implantado na Brasil Telecom e comparado com os dados históricos, ou melhor, com um possível contrato guarda-chuva. Não se tem a pretensão de afirmar que a solução por agentes seria mais eficiente, mas a possibilidade existe, segundo a percepção da própria equipe de Obtenção.

4.2.1 Especificação do Sistema Multiagente para Compra de Suprimentos de TI

Os compradores delegam a compra de suprimentos de TI solicitados pelos Núcleos a agentes de software. Este processo inclui a busca por fornecedores, a solicitação de cotações, decisão de qual fornecedor comprar (caso haja decisão de comprar), negociação com o fornecedor escolhido, para melhorar o acordo, e relatórios sobre o processo.

A Brasil Telecom, como uma grande compradora, pode impor algumas condições aos seus fornecedores de suprimento de TI, como ser detentora do “mercado eletrônico”, que também será um agente de software, e exigir que os fornecedores tenham também seus agentes de software para vender seus produtos. Estes agentes serão semelhantes aos agentes de compras, aos quais serão delegadas as condições de venda, bem como a negociação da mesma.

As variáveis de decisão para compras de suprimentos especificadas pela equipe de Obtenção da Brasil Telecom são:

- a) preço;
- b) prazo de entrega;
- c) quantidade;
- d) qualidade;
- e) prazo de pagamento,

f) marca.

Alguns comentários sobre as variáveis de compras mencionadas. Na Brasil Telecom o “prazo de pagamento” para tais tipos de compras é único. Geralmente, em torno de trinta dias, feitos em um único pagamento. É comum, entretanto, a mudança de tais prazos. Quem determina o prazo é a própria BrT, restando aos fornecedores aceitá-lo. Assim, a variável “prazo de pagamento” não entrará na negociação entre os agentes, ficando como uma variável restritiva.

Em relação à variável “marca”, ela também é restritiva. Caso sejam enumeradas as marcas que serão aceitas, todas as demais, por exclusão, não podem ser compradas.

A “qualidade” é, sem dúvida, a variável mais difícil de ser mensurada. Um produto pode ter uma qualidade alta para determinado comprador, mas baixa para outro, variando de suas experiências anteriores, adequação para seus propósitos, etc. Assim, a qualidade tem que ser dita pelo comprador.

A única variável de compra que é tradicionalmente lembrada e não consta na relação é “fornecedor”. Na verdade, ela está implícita. Assume-se que qualquer fornecedor participante do “mercado” já esteja habilitado para suprir os itens desejados. Eles são indiferentes em preferência.

A escolha do melhor fornecedor será realizada através do uso de funções de valor. Estas funções também possibilitarão o uso de algoritmos para a negociação colaborativa. As variáveis de decisão de compra são imprescindíveis para a criação da função de valor.

Serão três tipos de agentes: o Comprador, o Fornecedor e o Mercado. Cada um deles terá, resumidamente, as seguintes características:

a) Agente Mercado

Será um único agente, responsável por manter o cadastro dos Agentes Compradores e Agentes Fornecedores que estão ativos no mercado. Toda vez que um Agente Comprador solicitar os possíveis Agentes Fornecedores para o Agente Mercado, este último proverá uma lista com tais possíveis fornecedores.

O Agente Mercado terá o poder de decidir quem poderá entrar e continuar no mercado. Ele também será responsável pelo histórico das compras e pelas tentativas de compras, para controle do pessoal de Compras (no caso, Obtenção de TI ou DMS).

É o Agente Mercado que tem as especificações dos itens de compra, bem como a lista com os valores de qualidade de cada produto. Essas listas são determinadas pelo pessoal de compras, através de interface com o Agente Mercado.

b) Agente Comprador

Cada vez que se necessitar comprar um item, será criado um Agente Comprador, que fará interface, inicialmente, com o Comprador, neste caso, o próprio pessoal dos Núcleos. Com esta interação, o agente terá que receber a especificação do item e as algumas das variáveis de decisão. Também terá que montar a função de valor, de acordo com as preferências do Comprador e com as especificações de qualidade que o Agente Mercado dará.

Depois de dotado de todas as informações necessárias para compras, o Agente Comprador pedirá ao Agente Mercado uma lista com os possíveis fornecedores. De posse desta lista, enviará pedido de cotação de venda. Depois de recebidas as cotações, o Agente Comprador elegerá a melhor – aquela que tiver o maior retorno da função de valor. Eleito qual o melhor fornecedor, o agente iniciará um processo de negociação colaborativa (ganha-ganha) com o Agente Fornecedor.

Fechada a compra, o Agente Comprador informará ao Agente Mercado o histórico de todo o processo, para fim de controle das equipes de Obtenção e da Diretoria de Materiais e Serviços.

c) Agente Fornecedor

O Agente Fornecedor assume várias características duais do Agente Comprador. A fundamental diferença é que este agente não escolhe o melhor comprador. Contudo, a interação com o Fornecedor, captando suas preferências e montando uma função de valor, é de fundamental importância para viabilizar a negociação, em especial, a colaborativa.

4.2.2 Limitações

O sistema multiagente proposto terá algumas limitações, como:

- a) Restrição do universo de itens de compra – tudo foi pensado com os produtos da tabela do APÊNDICE A, como referência;
- b) Controle do Agente Mercado pela BrT;
- c) Impossibilidade dos Agentes Fornecedores de escolher o melhor comprador. Como simplificação, o fornecedor nunca “recusará” um cliente, se tiver condições de entregar o produto dentro do prazo exigido.

4.3 UMA SÍNTESE DOS PROCESSOS ATUAL E PROPOSTO

4.3.1 Atual

DMS e Obtenção de TI elaboram um contrato aberto. Quando um Núcleo solicita compra de um suprimento à Obtenção, esta última verifica se há previsão no contrato aberto. Se houver, encaminha um pedido para a DMS que autoriza a compra, de forma rápida, uma vez que há uma pré-autorização. Se não houver previsão, a Obtenção pesquisa os possíveis fornecedores e encaminha um pedido à DMS. Como não há previsão no contrato aberto, a DMS leva mais tempo para autorizar, pois precisa analisar o pedido de compra.

4.3.2 Proposto

A DMS, com ajuda da Obtenção, alimenta o Agente Mercado com informações dos produtos pré-autorizados. Quando um Núcleo necessitar de um suprimento, criará um agente Comprador que verificará, junto ao Agente Mercado, se há pré-autorização para a compra do suprimento em questão, bem como alguns outros parâmetros da compra. Se houver, o próprio agente será responsável pela compra. Caso não haja autorização, o Núcleo terá que usar o método atual.

Pelo proposto, haverá delegação de algumas das tarefas do processo de compras a agentes de software, que, como visto no Capítulo 2, surgiu exatamente para realizar autonomamente tarefas delegadas. Além da especificação do sistema visto neste capítulo, necessita-se de mais detalhes sobre o sistema, o que é o propósito do Capítulo 5.

O assunto do Capítulo 6 é como a decisão e a negociação de compras serão feitas. Fundamenta-se nos conceitos apresentados no terceiro capítulo: variáveis de decisão de compra, funções de valor e negociação colaborativa.

5 MODELANDO O SISTEMA MULTIAGENTE COM AUML

O Sistema Multiagente para Compras de Suprimentos de TI especificado no capítulo anterior será detalhado agora, através de diagramas, tabelas e textos explicativos. Este capítulo também objetiva discutir o uso da *Agent UML* como linguagem de modelagem.

Na próxima seção, serão apresentados os atores envolvidos no sistema especificado no Capítulo 4, bem como uma visão macro do relacionamento entre eles. Na seção 5.2, serão detalhadas as principais funcionalidades dos agentes. Os principais atributos necessários para uma futura implementação, estão descritos na seção 5.3. A seção 5.4 apresenta o conjunto de mensagens que os agentes dispõem para a comunicação entre os mesmos. Na 5.5, detalham-se as interações necessárias entre os agentes e entre estes e os usuários do sistema. Os principais fluxos de atividades realizados pelos agentes estão presentes na seção 5.6. Por fim, a seção 5.7 discute sobre mecanismos de decisão e negociação, apresentando uma solução própria para o sistema de compras de suprimentos de TI em questão.

5.1 DIAGRAMA DE CONTEXTO

No Diagrama de Contexto (Figura 7), é possível visualizar quais são os atores que vão interagir com o Sistema Multiagente para Compras de Suprimentos de TI. Os atores podem ser quaisquer entidades externas ao sistema, como pessoas, outros sistemas, etc. Para o sistema em questão, os atores são os Núcleos de TI, a Diretoria de Materiais e Serviços (DMS) e os fornecedores dos suprimentos.

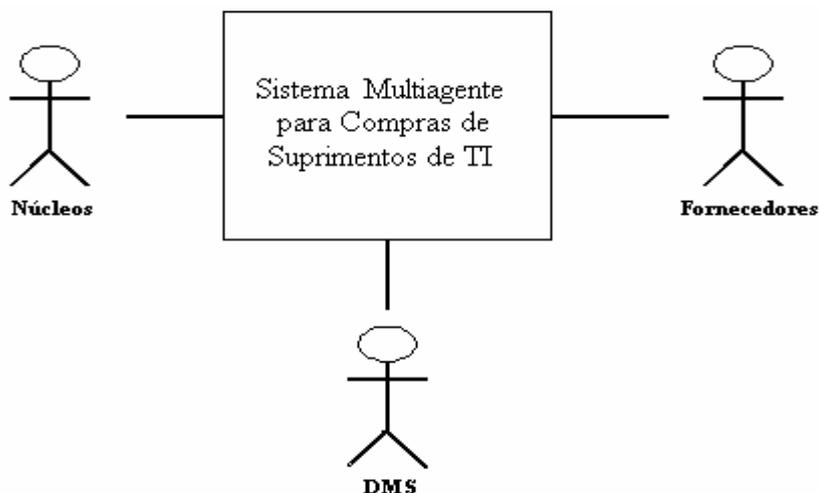


Figura 7 - Diagrama de Contexto

Este diagrama não faz parte da linguagem de modelagem UML, mas, como uma primeira visão do sistema, é bastante ilustrativo. Entretanto, o conceito de atores e a sua representação gráfica já estão sob as normas da UML.

5.2 DIAGRAMAS DE CASOS DE USO

Alguns conceitos são necessários para um bom entendimento dos Diagramas de Casos de Uso e para a discussão de como representar sistemas multiagentes por tais diagramas. Todos os conceitos foram retirados ou interpretados da especificação da UML (UML 1.4 Specification, 2002).

Um “caso de uso” é uma unidade funcional de um sistema, caracterizado por seqüências de mensagens trocadas entre o sistema e os atores e por um conjunto de ações realizadas por este sistema.

Um “ator” define um coerente conjunto de papéis que usuários de uma entidade podem desempenhar quando interagem com a mesma.

O Diagrama de Casos de Uso mostra atores e casos de uso juntos, com seus relacionamentos. Os casos de uso são representados por elipses e os atores, como já visto, como “bonecos”. Há alguns problemas no uso destes diagramas para modelar sistemas multiagentes. Talvez, o maior deles seja como representar um agente de software. Como casos de uso? Como atores?

Segundo a experiência de Cossentino, Chella e Lo Faso (2000), frequentemente, é melhor modelar agentes de software como casos de uso e, todas as outras entidades externas, como atores. Eles também acreditam que outros agentes, que se relacionam com um agente qualquer, devem ser modelados como atores.

Todo o raciocínio desses autores está baseado nos conceitos acima expostos e no seguinte conceito de agente: “Um agente é um sistema computacional encapsulado que está situado em algum ambiente e é capaz de agir de forma flexível e autônoma neste ambiente, de forma a atingir seus objetivos” (ALBUS, McCAIM, LUMIA apud COSENTINO, CHELLA e LO FASO, 2000).

Fazendo um paralelo entre este último conceito e o conceito de casos de uso anteriormente apresentado, os autores acreditam que um “sistema computacional encapsulado e que age sobre um ambiente” é uma “unidade funcional de um sistema” que se comunica e realiza ações. Já a interação com o ambiente, incluindo com outros agentes, está relacionada com as “seqüências de mensagens trocadas”.

Por outro lado, como a implementação de cada agente, em um sistema multiagente, independe das implementações dos demais, sendo necessário apenas a padronização da comunicação, um agente pode enxergar um outro como uma entidade externa, com o qual só se preocupa em como se comunicar, para atingir seus objetivos. Assim, quando se está modelando um agente (através de casos de uso), os outros podem ser vistos como atores.

Outros autores, como FLAKE, GEIGER e KÜSTER (2001), não optaram da mesma forma. Eles representaram seus agentes de software através de ambos: atores e casos de uso. Fizeram uma proposta de extensão da UML, onde há atores convencionais, atores que são agentes (representados com a cabeça quadrada), casos de usos (*use cases*), casos de objetivos (*goal cases*), casos de reação (*reaction cases*) e nuvens, que representam o ambiente.

Apesar da extensão torna-se interessante para a representação de agentes, mostrando as relações entre si e entre os agentes e o ambiente externo, houve um excesso de mudanças, constituindo um diagrama completamente novo, sendo pouco considerá-lo apenas uma extensão do diagrama de casos de uso da UML.

A diferenciação gráfica entre atores convencionais e atores que são agentes de software, entretanto, parece bastante útil para um bom entendimento do sistema, tanto por parte dos desenvolvedores do sistema, como para os usuários do mesmo.

Uma sugestão para representar sistemas multiagentes, através de Diagramas de Casos de Uso, é ter vários diagramas, onde se mostra cada agente, um por um, como casos de uso. Os demais ficam representados como atores, seguindo, porém, a notação da cabeça quadrada. Poder-se-ia ter um primeiro diagrama que traria todos os agentes como atores (Figura 8) e que, um por um, seriam “explodidos” em casos de uso, em próximos diagramas (Figura 9, Figura 10 e Figura 11).

Seguindo a sugestão acima, serão mostrados os diagramas para o Sistema Multiagente para Compras de Suprimentos de TI. O diagrama da Figura 8 traz o Diagrama de Contexto “explodido”, agora mostrando quais os componentes do sistema multiagente.

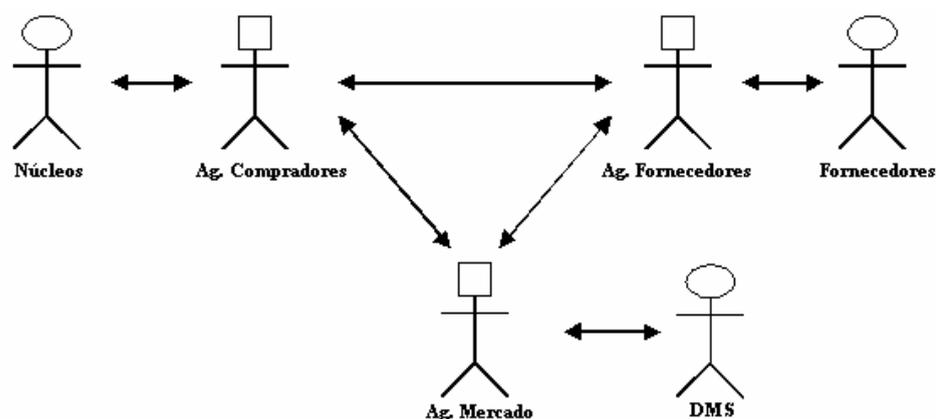


Figura 8 - Diagrama de Contexto Explodido em Atores

Nos próximos diagramas, cada agente será “explodido” em casos de uso, que irão se relacionar com um dos atores convencionais (Núcleos, Fornecedores ou DMS) e com os dois demais atores agentes.

O primeiro tipo de agente a ser “explodido” será o Agente Comprador. Note, no diagrama seguinte, que o retângulo reúne vários casos de uso, que, juntos, representam um Agente Comprador. Podem existir vários Agentes Compradores e Agentes Fornecedores. Assim, decidiu-se manter nos diagramas a nomenclatura no plural. Muitas vezes, contudo, pelo contexto, será percebido que se fala em uma instância destas “classes” de agentes, ou

seja, estará se mostrando como um único agente se comporta, mas o recorrido servirá para quaisquer agentes dessas “classes”. Outras vezes, fala-se sobre algumas ou todas as instâncias.

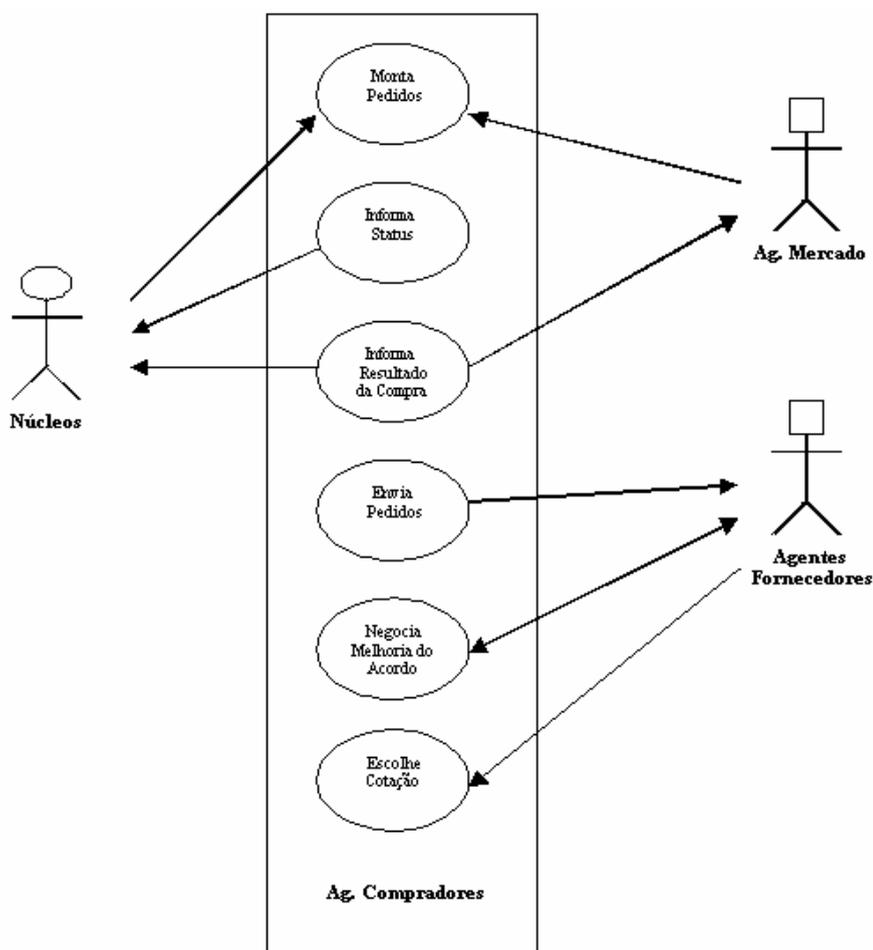


Figura 9 - Diagrama de Casos de Uso (Explosão dos Agentes Compradores)

Serão usadas tabelas para dar detalhes de cada um dos casos de uso.

Tabela 1 - Caso de Uso: Monta Pedido

Nome	Monta Pedidos
Atores envolvidos	Núcleos e Ag. Mercado.
Outros Casos de Uso referenciados	-
Precondições	O Agente Comprador já ter sido criado.
Iniciação	✓ O Núcleo informa os requisitos de compra (produto, unidade, quantidade, quantidade máxima e prazo máximo de entrega, núcleo solicitante, técnico solicitante).
Descrição da execução	<p>✓ Preencher requisitos de compra, informados pelo Núcleo solicitante e pelo Agente Mercado (prazo de pagamento, preço máximo, “qualidade” de cada uma das marcas, possíveis fornecedores, marcas aceitáveis);</p> <p>✓ Criar estrutura de preferências.</p>
Conclusão	✓ Formatar um pedido de cotação para ser enviado aos possíveis fornecedores (dependendo da implementação, um pedido pode ser uma estrutura única ou, simplesmente, uma verificação se todos os atributos necessários para fazer um pedido foram preenchidos).

Tabela 2 - Caso de Uso: Informa Status

Nome	Informa Status
Atores envolvidos	Núcleos.
Outros Casos de Uso referenciados	-
Precondições	-
Iniciação	✓ Núcleo solicita informações sobre o Status do Agente Comprador.
Descrição da execução	✓ Avalia o Status do Agente Comprador, através de suas propriedades.
Conclusão	✓ Retorna o Status do Agente Comprador: montando pedido de cotação, esperando cotações, negociando melhoria ou compra terminada.

Tabela 3 - Caso de Uso: Informa Resultado da Compra

Nome	Informa Resultado da Compra
Atores envolvidos	Agente Mercado e Núcleos.
Outros Casos de Uso referenciados	Monta Pedidos, Escolhe Cotação e Negocia Melhoria do Acordo.
Precondições	Não ter nenhum Fornecedor que atenda aos requisitos de compra ou já ter ocorrido a negociação de melhoria do acordo.
Iniciação	✓ Normalmente, depois do término do caso de uso “Negocia Melhoria do Acordo”; ✓ Em caso de não ter sido possível a compra, depois do término dos casos de uso “Monta Pedidos” ou “Escolhe Cotação”.
Descrição da execução	✓ É a última etapa feita pelo Agente Comprador, onde reporta ao Núcleo solicitante e ao Agente Mercado como se deu o processo de compra, se houve êxito ou não, com detalhes.
Conclusão	✓ Após informar os resultados, o criador do agente (Núcleo) libera-o.

Tabela 4 - Caso de Uso: Envia Pedidos

Nome	Envia Pedidos
Atores envolvidos	Agentes Fornecedores.
Outros Casos de Uso referenciados	Monta Pedidos.
Precondições	Existência de um pedido de cotação formatado (ou, todos os atributos necessários tenham sido preenchidos, dependendo da implementação).
Iniciação	✓ É iniciado após o término do caso de uso “Monta Pedidos”, se o mesmo formatar um pedido de cotação.
Descrição da execução	✓ Recebe pedido de cotação formatado; ✓ Cria mensagens para os Agentes Fornecedores que estão na lista “possíveis fornecedores”, enviadas pelo Agente Mercado, contendo os Pedidos de Cotação.
Conclusão	✓ Registrar a mensagem e os Agentes Fornecedores para os quais enviou a mesma, a fim de controle posterior da DMS.

Tabela 5 - Caso de Uso: Negocia Melhoria do Acordo

Nome	Negocia Melhoria do Acordo
Atores envolvidos	Agentes Fornecedores.
Outros Casos de Uso referenciados	Escolhe Cotação.
Precondições	Haver um Agente Fornecedor que tenha sido escolhido no caso de uso “Escolhe Cotação”.
Iniciação	✓ O Agente Comprador convida o Agente Fornecedor escolhido para negociar melhoria;
Descrição da execução	✓ Faz o “handshake”, que é enviar alguns limites de suas funções, a fim de tornar compatíveis as funções dos negociadores; ✓ O Agente Comprador envia sucessivas propostas de melhoria. Em cada proposta, há um conjunto de acordos possíveis indiferentes em preferência, para o Fornecedor escolher o que maximize sua função, se houver alguma; ✓ Se o Ag. Fornecedor aceitar algum novo acordo, esse passa a valer como o

	<ul style="list-style-type: none"> atual; ✓ Os papéis são invertidos, de forma que o Ag. Comprador aguarda as propostas do Ag. Fornecedor, de forma dual ao feito por ele; ✓ Se o Ag. Comprador aceitar algum acordo de alguma proposta, esse passa a valer como o atual; ✓ O Ag. Comprador volta a enviar sucessivas propostas de melhoria, em um ciclo alternado, que será repetido até que a condição de parada do algoritmo implementado seja satisfeita.
Conclusão	✓ Quando a condição de parada do algoritmo implementado for satisfeita.

Tabela 6 - Caso de Uso: Escolhe Cotação

Nome	Escolhe Cotação
Atores envolvidos	Agentes Fornecedores.
Outros Casos de Uso referenciados	Envia Pedidos de Cotação.
Precondições	Os pedidos de cotação já terem sido enviados.
Iniciação	✓ Inicia-se assim que termina o caso de uso “Envia Pedidos de Cotação”.
Descrição da execução	<ul style="list-style-type: none"> ✓ Espera pela chegada de todos os pedidos de cotação ou pelo término do prazo (<i>deadline</i>) de espera; ✓ Calcula a função valor total de cada uma das cotações recebidas; ✓ Ordena as cotações, sendo a melhor opção aquela com o maior retorno da função valor total.
Conclusão	✓ Envia mensagem para o Agente Fornecedor que ganhou a concorrência, podendo o Agente Comprador dar início à negociação de melhoria.

O Agente Fornecedor tem algumas semelhanças com o Agente Comprador, como montar estruturas de preferências ou a capacidade de negociar melhorias. Outras, como montar venda e enviar cotações, são inerentes a este tipo de agente. O seguinte diagrama mostra a “explosão” dos Agentes Fornecedores em casos de uso.

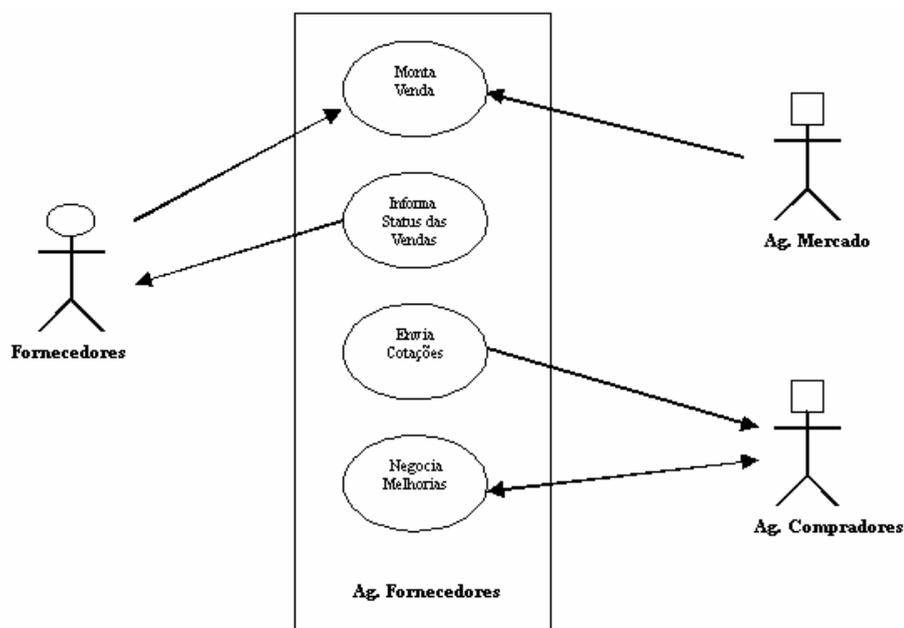


Figura 10 - Diagrama de Casos de Uso (Explosão dos Agentes Fornecedores)

Detalhando os casos de uso:

Tabela 7 - Caso de Uso: Monta Venda

Nome	Monta Venda
Atores envolvidos	Fornecedores e Ag. Mercado.
Outros Casos de Uso referenciados	-
Precondições	O Agente Fornecedor já ter sido criado.
Iniciação	✓ O Fornecedor informa os requisitos de venda (nome do fornecedor, nome do criador do agente, produto, unidade, marca, quantidade, prazos mínimo e ideal de entrega, prazo máximo para recebimento).
Descrição da execução	<ul style="list-style-type: none"> ✓ Preencher requisitos de compra informados pelo Fornecedor; ✓ Criar estrutura de preferências; ✓ Registrar o Agente Fornecedor no “mercado”, ou seja, juntamente ao Agente Mercado.
Conclusão	✓ Registro aceito ou negado pelo Ag. Mercado.

Tabela 8 - Caso de Uso: Informa Status das Vendas

Nome	Informa Status das Vendas
Atores envolvidos	Fornecedores.
Outros Casos de Uso referenciados	-
Precondições	-
Iniciação	✓ Fornecedor solicita informações sobre o Status do Agente.
Descrição da execução	✓ Avalia as vendas e tentativas de venda realizadas até o momento.
Conclusão	✓ Retorna informações sobre as vendas realizadas, bem como informações sobre todas as cotações enviadas, incluindo aquelas em que espera resposta ou que não tenham sido escolhidas.

Tabela 9 - Caso de Uso: Envia Cotação

Nome	Envia Cotações
Atores envolvidos	Agentes Compradores.
Outros Casos de Uso referenciados	-
Precondições	Estar cadastrado pelo Agente Mercado.
Iniciação	✓ Recebe pedido de cotação de um Agente Comprador.
Descrição da execução	✓ Após receber pedido de cotação, avalia se as condições impostas pelo Agente Comprador são preenchidas.
Conclusão	✓ Envio de cotação, em caso de preencher os requisitos do pedido.

Tabela 10 - Caso de Uso: Negocia Melhorias

Nome	Negocia Melhorias
Atores envolvidos	Agentes Compradores.
Outros Casos de Uso referenciados	-
Precondições	Ter enviado cotação para algum Agente Comprador.
Iniciação	✓ Algum Agente Comprador envia uma mensagem, comunicando o Agente Fornecedor que ele foi o escolhido.
Descrição da execução	<ul style="list-style-type: none"> ✓ Faz o “handshake”, definindo alguns dos limites de suas funções; ✓ De forma dual, segue o ciclo informado no Caso de Uso Negocia Melhoria do Acordo, visto para o Ag. Comprador.
Conclusão	✓ Quando a condição de parada do algoritmo implementado for satisfeita.

Como última “explosão” de um agente em casos de uso, tem-se o Agente Mercado no diagrama da Figura 11:

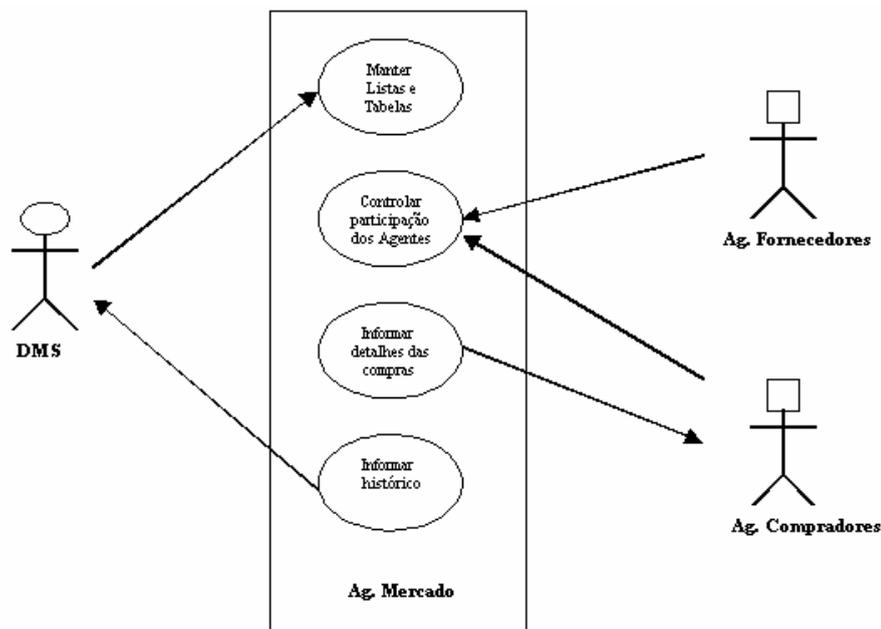


Figura 11 - Diagrama de Casos de Uso (Explosão do Agente Mercado)

Seguem os detalhes dos casos de uso:

Tabela 11 - Caso de Uso: Manter Listas e Tabelas

Nome	Manter Listas e Tabelas
Atores envolvidos	DMS.
Outros Casos de Uso referenciados	-
Precondições	-
Iniciação	-
Descrição da execução	✓ O Agente Mercado é responsável pela manutenção de várias listas e/ou tabelas, através de inclusões, exclusões, alterações e consultas às mesmas. O único ator que interage diretamente com estas informações é a DMS. Os demais só indiretamente, através de decisões do próprio Agente Mercado.
Conclusão	-

Tabela 12 - Caso de Uso: Controlar Participação dos Agentes

Nome	Controlar participação dos Agentes
Atores envolvidos	Agentes Compradores e Agentes Fornecedores.
Outros Casos de Uso referenciados	-
Precondições	-
Iniciação	-
Descrição da execução	<p>✓ Caso esteja se relacionando com um Ag. Fornecedor, o que o Ag. Mercado tem que fazer é simplesmente verificar se este agente é representante de algum Fornecedor autorizado pela DMS, consultando a lista de fornecedores autorizados. Em caso positivo, registra o Ag. Fornecedor, incluindo o produto que está vendendo, em uma lista dos Agentes Fornecedores ativos no mercado. Em caso negativo, não permite o registro do Ag. Fornecedor. Também há a possibilidade de retirar algum Ag. Fornecedor do mercado, sempre que a DMS desejar não mais negociar com o fornecedor que o agente representa;</p> <p>✓ Se estiver relacionando-se com um Agente Comprador, verifica se o produto solicitado está previsto e pré-autorizado para compra. Se estiver, registra o Ag. Comprador no “mercado”, caso contrário, avisa ao agente qual o problema da não autorização (produto não previsto, quantidade superior à autorizada, etc.).</p>
Conclusão	-

Tabela 13 - Caso de Uso: Informar Detalhes das Compras

Nome	Informar Detalhes das Compras
Atores envolvidos	Agentes Compradores.
Outros Casos de Uso referenciados	
Precondições	O Agente Comprador ter sido registrado no “mercado”.
Iniciação	-
Descrição da execução	✓ Para cada Agente Comprador, o Agente Mercado informa detalhes da compra (preço máximo a ser pago, prazo de pagamento, “qualidade” das marcas e possíveis fornecedores).
Conclusão	-

Tabela 14 - Caso de Uso: Informar Histórico

Nome	Informar Histórico
Atores envolvidos	DMS.
Outros Casos de Uso referenciados	-
Precondições	-
Iniciação	-
Descrição da execução	✓ É responsabilidade do Agente Mercado, informar à DMS sobre as movimentações do mercado durante qualquer período desejado. Estas informações são imprescindíveis para um controle da DMS sobre as compras realizadas pelos Núcleos.
Conclusão	-

5.3 ATRIBUTOS DOS AGENTES

Alguns atributos são importantes para um bom entendimento de como funcionará o sistema multiagente. Dependendo da implementação, estes atributos podem assumir diversas formas e nomes, mas sempre existirão. Nas tabelas seguintes, serão especificados os principais atributos (e suas descrições) dos agentes.

Tabela 15 - Principais atributos do Agente Mercado

Atributo	Descrição
FornecedoresAutorizados[]	Lista dos fornecedores autorizados a participar do mercado. A definição desta lista é feita pela DMS.
ProdutosAutorizados[]	Tabela com a relação dos produtos pré-autorizados pela DMS para compra, contendo a descrição destes produtos, as marcas possíveis, o preço unitário máximo e quantidade máxima (abatida a cada autorização de compra e acrescida somente pela DMS). O APÊNDICE A traz uma tabela parecida com este atributo. Por exemplo, o produto “Mouse PS/2” só será aceito das marcas Microsoft, Logitech ou Genius. Seu preço máximo para compra é R\$12,85. Há uma pré-autorização para comprar até 110 unidades.
AgentesCompradoresAtuais[]	Lista contendo os atuais Agentes Compradores presentes no mercado.
AgentesFornecedores[]	Tabela com os agentes fornecedores cadastrados (ativos) no mercado e, para cada um deles, o seu produto ofertado.
qualidadeDasMarcas[]	Tabela produto × marca × qualidade. Através desta tabela, a DMS informa qual a qualidade percebida pela empresa sobre os produtos de determinadas marcas. Voltando ao exemplo do “Mouse PS/2”, poder-se-ia ter uma tabela que informa que o mouse de marca Microsoft tem “qualidade” 9, o Logitech, 8, e o Genius, 7, por exemplo.
historicoCompras[]	Tabela que armazena informações sobre as compras e as tentativas de compra feitas pelos Agentes Compradores. Essas informações são organizadas no tempo, de forma que garantam relatórios por períodos para a DMS.

Tabela 16 - Principais atributos do Agente Comprador

Atributo	Descrição
agenteCompradorID	Identificador único do Agente Comprador, dado pelo Agente Mercado no momento do registro.
nucleoSolicitante	Nome do Núcleo que criou o agente e está solicitando a compra.
tecnicoSolicitante	Nome de pessoa do Núcleo que está interagindo com o sistema.
produto	Descrição do produto a ser comprado.
unidade	Qual a unidade deste produto: metro, caixas, quilograma, unidade, etc.
quantidadeMaxima	A máxima quantidade que o Núcleo pode aceitar.
quantidadeMinima	A quantidade estimada como realmente necessária pelo núcleo. Geralmente, só esta quantidade é especificada ou, ao menos, não há necessidade das demais, sendo esta a ideal, mínima e máxima.
prazoMaximoEntrega	O prazo máximo para que o produto comprado seja entregue.
prazoPagamento	O prazo de pagamento informado pelo Agente Mercado.
possiveisFornecedores[]	Lista dos possíveis fornecedores para o produto desejado, informada pelo Agente Mercado.
marcasEQualidade[]	Tabela das marcas aceitas com as respectivas “notas” de qualidade.
precoMaximo	Preço máximo, por unidade, autorizado pela DMS.
funcaoValorTotal	Descrição da estrutura de preferências (deve ser implementada como uma estrutura composta – uma soma ponderada de funções).
historicoNegociacao[]	Lista com informações sobre o histórico da negociação, incluindo as propostas e os números que a identificam.
cotacoesRecebidas[]	Lista contendo todas as cotações recebidas.
deadlineCotacoes	Prazo máximo de espera por cotações.
statusCompra	Uma variável que diz o status atual do processo de compras, podendo assumir um dos seguintes valores: montandoPedido esperandoCotacoes negociandoMelhorias compraFinalizada.
previsaoEntrega	Informa a previsão de entrega do produto, quando o Agente Comprador está no status “compra finalizada”.

Tabela 17 - Principais atributos do Agente Fornecedor

Atributo	Descrição
AgenteFornecedorID	Identificador único do agente fornecedor, dado pelo Agente Mercado no momento do registro.
fornecedorRepresentado	Nome do fornecedor que o Agente Fornecedor está representando.
criadorDoAgFornecedor	Nome da pessoa que interagiu com o sistema e é responsável pelo agente.
produto	Descrição do produto a ser vendido, por exemplo, “Mouse OS/2”.
marca	Marca do produto ofertado, por exemplo, “Genius”.
quantidade	Quantidade disponível para venda.
unidade	A unidade do produto (metro, caixas, unidade, quilograma, etc.).
precoMinimo	Preço unitário mínimo de venda.
preco	Preço que será enviado nas cotações.
prazoEntregaMinimo	Prazo mínimo possível para que o Fornecedor entregue algum pedido.
prazoEntrega	Prazo para entrega do produto ofertado, enviado na cotação.
prazoMaximoRecebimento	Prazo máximo que o Fornecedor está disposto a esperar pelo pagamento da venda.
pedidosDeCotacaoRecebidos[]	Listas com os pedidos de cotação recebidos.
cotacoesEnviadas[]	Lista com as cotações enviadas para os Agentes Compradores.
agentesEmNegociacao[]	Lista com todos os Agentes Compradores que estão em negociação no momento, incluindo o histórico da negociação.
vendasRealizadas[]	Lista com o registro das vendas realizadas.
listaFuncoesValorTotal	Lista contendo as descrições das estruturas de preferência (deve ser implementada como uma estrutura composta – uma soma ponderada de funções), uma para cada Ag. Comprador em negociação.
situacaoRegistro	Descreve a situação do registro do Agente Fornecedor no Agente Mercado, podendo assumir os seguintes valores: naoSolicitado liberado negado.
statusDoAgente	Especifica qual o status atual do Agente Fornecedor: montandoVenda registrandoNoMercado ativoNoMercado.

Para que sejam implementados, os agentes ainda necessitarão de muitos outros atributos, que surgirão com as necessidades encontradas durante as implementações. As listas e tabelas podem assumir diversas formas, quando implementadas. Os detalhes de implementação não são o foco deste trabalho.

5.4 CONJUNTO DE MENSAGENS

Na próxima seção serão mostradas as interações entre os atores do sistema. O primeiro passo, entretanto, é definir um conjunto de mensagens, para que seja possível a comunicação. A Tabela 18 traz este conjunto, especificando as mensagens, suas descrições, de onde partem, para onde se dirigem e quais os parâmetros das mesmas. Para facilitar a leitura posterior dos diagramas de interação, as mensagens estão dispostas em uma ordem semelhante as suas aparições nestes diagramas.

Tabela 18 - Conjunto de Mensagens Trocadas entre os Atores

MENSAGEM	DESCRIÇÃO	ORIGEM	DESTINO	PARÂMETROS
InformaRequisitosCompra() [Figura 12]	O solicitante do Núcleo informa os requisitos básicos da compra, como a especificação do produto, quantidades máxima e mínima, etc.	Núcleo	Ag. Comprador	- núcleo solicitante - técnico solicitante - produto - unidade - quantidades: máxima e mínima - prazo máximo de entrega
SolicitaQuestionarioCompra() [Figura 12]	O Ag. Comprador solicita ao Núcleo informações sobre suas preferências. Pode ser, por exemplo, um conjunto de funções e os pesos associados às mesmas.	Ag. Comprador	Núcleo	- questionário de preferências
RespondeQuestionarioCompra() [Figura 12]	O técnico solicitante do Núcleo envia resposta do questionário para o Ag. Comprador. Nesse questionário, serão inferidas as preferências do Comprador.	Núcleo	Ag. Comprador	- questionário respondido
SolicitaStatusCompra() [Figura 12]	Núcleo solicita do Ag. Comprador informações sobre o status da compra.	Núcleo	Ag. Comprador	-
InformaStatusCompra() [Figura 12]	Ag. Comprador informa o status da compra ao Núcleo. Se a compra já foi finalizada, informa também a previsão de chegada.	Ag. Comprador	Núcleo	- status da compra (recebendo cotações/negociando melhoria/finalizada, previsão de entrega)
InformaResumoCompra() [Figura 12]	O Ag. Comprador informa ao Comprador o desfecho da compra.	Ag. Comprador	Núcleo	- houve compra? Sim/não - produto - marca - preço por unidade - quantidade - prazo de entrega - fornecedor
LiberaAgente() [Figura 12;Figura 13]	Utilizado por quaisquer dos atores (Compradores, Fornecedores ou Mercado) para “destruir” seus agentes, liberando memória.	Núcleos, Fornecedores ou DMS	Ag. Compradores, Ag. Fornecedores ou Ag. Mercado.	- identificação do agente

InformaRequisitosVenda() [Figura 13]	Dual, para o Fornecedor, ao InformaRequisitoCompras().	Fornecedor	Ag. Fornecedor	- nome do fornecedor - nome do criador do agente - produto - unidade - marca - quantidade disponível - prazos mínimo e ideal de entrega - prazo máximo para recebimento do valor da venda
SolicitaQuestionarioVenda() [Figura 13]	O Ag. Fornecedor solicita ao Fornecedor informações sobre suas preferências.	Ag. Fornecedor	Fornecedor	- questionário de preferências
RespondeQuestionarioVenda() [Figura 13]	Dual, para o Fornecedor, ao RespondeQuestionarioCompra().	Fornecedor	Ag. Fornecedor	- questionário respondido
SolicitaStatusVenda() [Figura 13]	Fornecedor solicita do Ag. Fornecedor informações sobre o status da Venda.	Fornecedor	Ag. Fornecedor	-
InformaStatusVenda() [Figura 13]	Ag. Fornecedor informa o status da venda ao Fornecedor.	Ag. Fornecedor	Fornecedor	- status da Venda (listas das cotações solicitadas e enviadas, listas das vendas efetuadas até agora, em detalhes, lista dos agentes com quem está negociando no momento)
AumentaQuantidade() [Figura 13]	Fornecedor informa ao Ag. Fornecedor o aumento no estoque do produto à venda.	Fornecedor	Ag. Fornecedor	- quantidade acrescida
ConfirmaAumentoQuantidade() [Figura 13]	Ag. Fornecedor informa ao Fornecedor que processou o aumento na quantidade ofertada.	Ag. Fornecedor	Fornecedor	-
InformaResumoParcialVenda() [Figura 13]	O Agente Fornecedor informa ao Fornecedor toda venda feita para cada Agente Comprador.	Ag. Fornecedor	Fornecedor	- identificação do Agente Comprador - produto - marca - preço por unidade - quantidade - prazo de entrega - prazo de pagamento (quando vai receber) - histórico da negociação colaborativa
OperaListasOuTabelas() [Figura 14]	São mensagens enviadas pela DMS para que o Agente Mercado realize algumas operações nas listas e/ou tabelas, como incluir um novo fornecedor autorizado ou consultar quais os atuais agentes presentes no mercado.	DMS	Ag. Mercado	- tipo da operação: inclusão, exclusão, alteração ou consulta - lista ou tabela - "informação"
SolicitaResumoMercado() [Figura 14]	A DMS solicita um resumo das atividades do mercado no período estipulado.	DMS	Ag. Mercado	- período
InformaResumoMercado() [Figura 14]	Ag. Mercado envia o resumo das atividades do mercado no período solicitado.	Ag. Mercado	DMS	- histórico do período
RegistraAgenteFornecedor() [Figura 15]	Usada para um Ag. Fornecedor entrar no mercado (cadastrando-se no Ag. Mercado), informando o produto que tem para vender e nome do Fornecedor que representa.	Ag. Fornecedor	Ag. Mercado	- Produto que tem para vender - nome do Fornecedor
AutorizaFornecedor() [Figura 15]	Mensagem que o Ag. Mercado dá ao Ag. Fornecedor, autorizando ou não sua entrada no Mercado. Em caso de autorização, informa seu número de identificação.	Ag. Mercado	Ag. Fornecedor	- autorizado? Sim/não - número de identificação no mercado
InformaSaidaMercado() [Figura 15;Figura 16]	O Ag. Comprador ou Ag. Fornecedor informa ao Ag. Mercado sua saída do mercado.	Ag. Comprador ou Ag. Fornecedor	Ag. Mercado	- identificação

RegistraAgenteComprador() [Figura 16]	Usada para um Ag. Comprador entrar no mercado (cadastrando-se no Ag. Mercado), informando o produto desejado, quantidade desejada e o nome do Comprador (técnico do núcleo que está interagindo com o sistema) que representa.	Ag. Comprador	Ag. Mercado	- Produto desejado - Quantidade - Nome do Comprador
AutorizaComprador() [Figura 16]	Mensagem que o Ag. Mercado dá ao Ag. Comprador, autorizando ou não sua entrada no Mercado. Em caso de autorização, informa uma lista de possíveis fornecedores, além de um número de identificação para o Ag. Comprador. Também informa os pré-requisitos da compra, como preço máximo, prazo de pagamento, marcas aceitáveis e a tabela: marcas X qualidade.	Ag. Mercado	Ag. Comprador	- autorizado? Sim/não - número de identificação do Agente Comprador no mercado - lista de possíveis fornecedores - preço máximo - prazo de pagamento - listas das possíveis marcas tabela: marcas X qualidade
InformaFinalCompra() [Figura 16]	O Ag. Comprador informa ao Ag. Mercado o final da compra, detalhando a transação.	Ag. Comprador	Ag. Mercado	- identificação própria - houve compra? Sim/não - identificação do agente fornecedor (em caso de ter havido compra) - produto - marca - quantidade comprada - preço unitário - prazo de entrega - prazo de pagamento - estrutura de preferências - lista de todas as cotações recebidas - histórico da negociação
SolicitaCotacao() [Figura 17]	Ag. Comprador envia, para cada um dos possíveis Agentes Fornecedores, solicitações de cotação.	Ag. Comprador	Ag. Fornecedores	- identificação - produto desejado - marcas possíveis - quantidade desejada - prazo de pagamento
EnviaCotacao() [Figura 17]	Ag. Fornecedor envia cotação solicitada do produto pelo Ag. Comprador.	Ag. Fornecedor	Ag. Comprador	- identificação - cotação (preço unitário, quantidade, prazo de entrega, marca)
ComunicaEscolhido() [Figura 17]	O Ag. Comprador comunica ao Ag. Fornecedor de melhor cotação que o mesmo fora escolhido – isso quando há alguma cotação que atenda aos requisitos.	Ag. Comprador	Ag. Fornecedor	- identificação
HandShake() [Figura 17]	O Ag. Comprador envia, para o Ag. Fornecedor, informações sobre os limites máximos de suas funções, a fim de torná-las compatíveis às funções do Ag. Fornecedor.	Ag. Comprador	Ag. Fornecedor	- limites máximos (preço, prazo e quantidade) das funções
EnviaProposta() [Figura 17]	O Ag. Comprador ou o Ag. Fornecedor envia uma proposta para seu parceiro de negociação, contendo um conjunto de acordos possíveis e indiferentes em preferência.	Ag. Comprador ou Ag. Fornecedor	Ag. Comprador ou Ag. Fornecedor	- identificação - proposta - número da proposta de melhoria
RespondeProposta() [Figura 17]	O agente que recebeu a proposta responde se aceita alguns dos possíveis acordos.	Ag. Fornecedor ou Ag. Comprador	Ag. Fornecedor ou Ag. Comprador	- sim não - se sim, o novo acordo
ConfirmaCompra() [Figura 17]	O Ag. Comprador confirma a compra ao Ag. Fornecedor com quem negociou, informando como ficaram as variáveis de compra.	Ag. Comprador	Ag. Fornecedor	- identificação - confirmado? Sim/não - produto - marca - preço unitário - quantidade - prazo de entrega - prazo de pagamento

5.5 DIAGRAMAS DE INTERAÇÃO

Os diagramas de interação servem para explicitar o comportamento dinâmico do sistema modelado. A UML especifica dois tipos de diagramas de interação, o Diagrama de Seqüência e o Diagrama de Colaboração. O primeiro deles enfatiza a ordem em que os eventos ocorrem, já o seguinte, a estrutura dos objetos. Contudo, os dois são equivalentes na semântica, diferem apenas no formato, ficando a cargo do modelador qual ele acha mais interessante para representar o que deseja.

Escolheu-se, para a representação deste sistema, utilizar apenas os diagramas de seqüência, por terem sido considerados os mais adequados, além de terem sido encontrados mais facilmente na literatura revisada.

Na Figura 8, que mostra os atores convencionais e os atores agentes, não fica claro como se dará o relacionamento entre estes atores. Para tanto, serão usados os Diagramas de Seqüência, detalhando como são essas interações.

Serão apresentados agora, um por um, os diagramas de seqüência. O primeiro deles mostrará a interação entre os Núcleos e os Agentes Compradores.

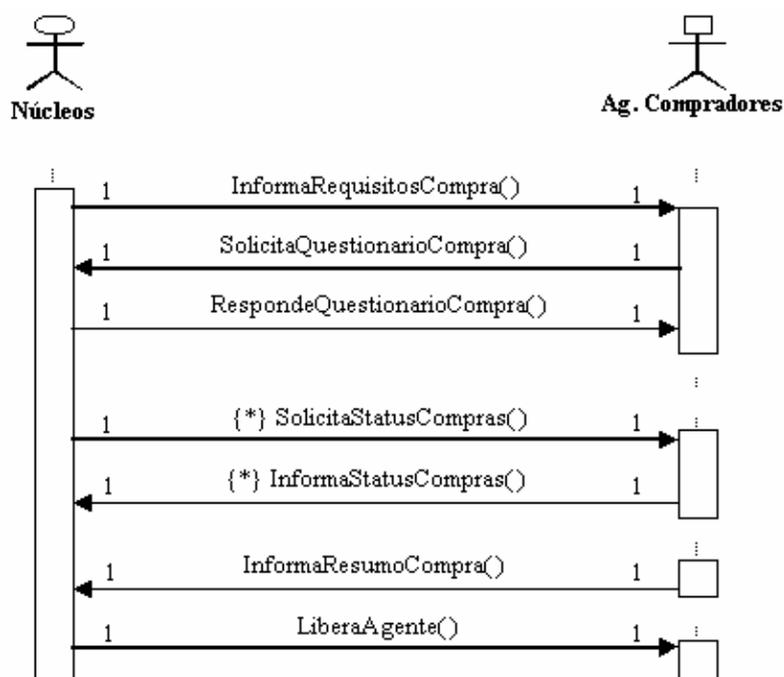


Figura 12 - Diagrama de Seqüência: Núcleos e Ag. Compradores

Como pode ser visto na Figura 12, a primeira interação entre o Núcleo e o Agente Comprador é informar quais são os requisitos da compra. O próximo passo é dado pelo Agente Comprador que envia um questionário para inferir as preferências do solicitante, que responde, sendo esta etapa imprescindível para a criação da função de valor.

Os numerais “1”, vistos no diagrama, especificam a cardinalidade da relação. Como podem ser observadas, todas as relações são um-para-um. Isto é devido ao fato de, a cada compra que o Núcleo deseja realizar (por produto), é criado um agente de software para delegar esta função.

As interações “SolicitaStatusCompra()” e “InformaStatusCompras()” estão marcadas pelos símbolos “{*}” para informar que podem ser feitas várias vezes, a qualquer momento. As demais, só acontecem uma vez durante toda a “vida” do agente.

Ao terminar o processo de compra, o Agente Comprador informará um resumo do processo para o solicitante (Núcleo), que, em seguida, libera o agente.

Nos diagramas de seqüência, as linhas pontilhadas verticais indicam a linha de vida das entidades. O tempo aumenta de cima para baixo na linha. Note-se também que há retângulos finos acompanhando as linhas pontilhadas. A continuidade do retângulo significa que o ator em questão está processando, exclusivamente, a resposta para a mensagem recebida. Como o Núcleo é um ator externo – não faz parte do sistema – representam-no como um retângulo contínuo, do início ao fim da sua linha de vida. Já o Agente Comprador divide suas atenções entre as interações com o Núcleo e com os outros dois agentes. Assim, só em alguns momentos, seu processamento pára, aguardando uma resposta a alguma mensagem. Esta discussão é válida, por analogia, para todos os outros Diagramas de Seqüência.

O próximo diagrama traz as interações entre os Fornecedores e seus Agentes Fornecedores.

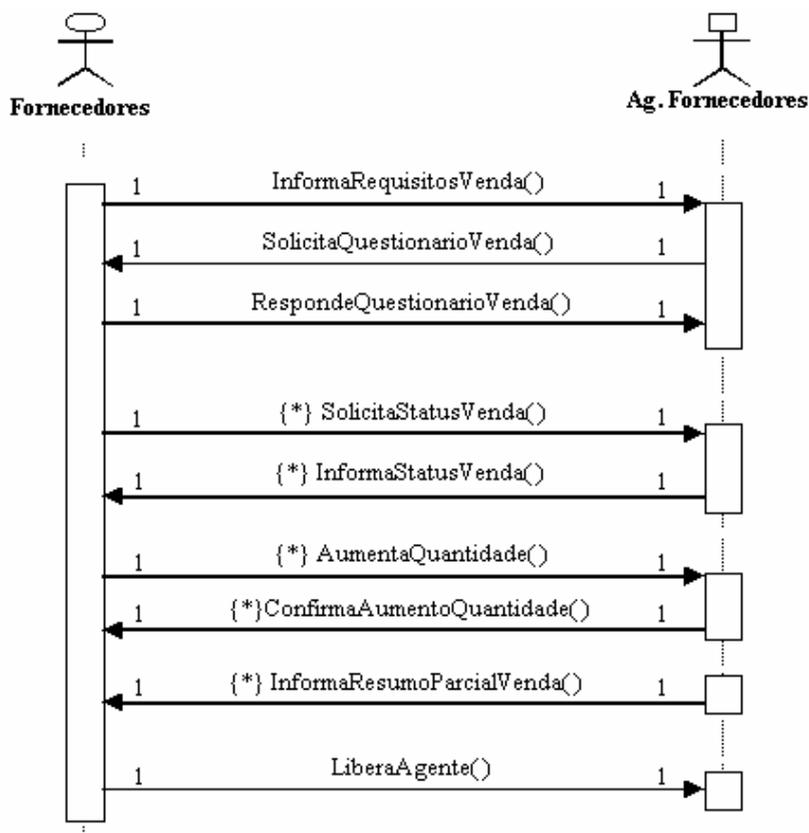


Figura 13- Diagrama de Seqüência: Fornecedores e Ag. Fornecedores

As relações no diagrama acima também são todas da forma um-para-um, também porque cada Fornecedor cria um agente para vender determinado produto. As interações que estão marcadas com o símbolo “{ }” – lembrando – significa que podem ser realizadas quantas vezes forem desejadas, bem como, a qualquer momento.

As interações entre a Diretoria de Materiais e Serviços (DMS) e o Agente Mercado são mostradas a seguir:

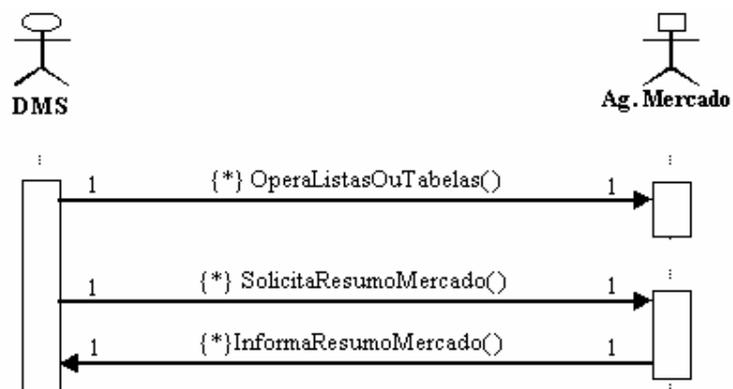


Figura 14- Diagrama de Seqüência: DMS e Ag. Mercado

Entre os Agentes Fornecedores e o Agente Mercado, têm-se as seguintes interações:

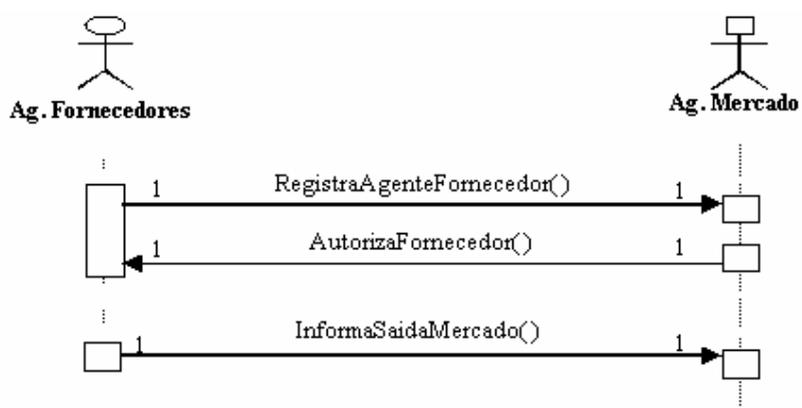


Figura 15- Diagrama de Seqüência: Ag. Fornecedores e Ag. Mercado

O próximo digrama mostra como se relacionam os Agentes Compradores e o Agente Mercado:

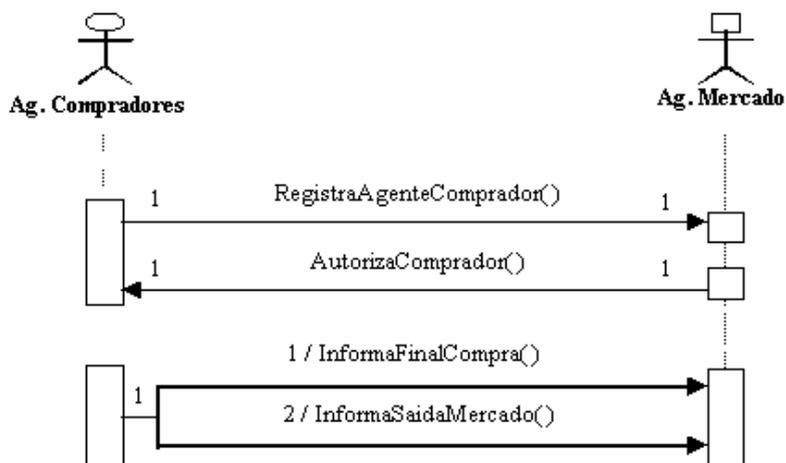


Figura 16- Diagrama de Seqüência: Ag. Comprador e Ag. Mercado

Observa-se no diagrama da Figura 16 que há duas mensagens paralelas: “InformaFinalCompra()” e “InformaSaidaMercado()”. Apesar de paralelas, elas obedecem à ordem definida pelos numerais à esquerda, que estão sucedidos pelo símbolo “/”.

Por fim, há um importante diagrama que mostra as interações entre os Agentes Compradores e Fornecedores:

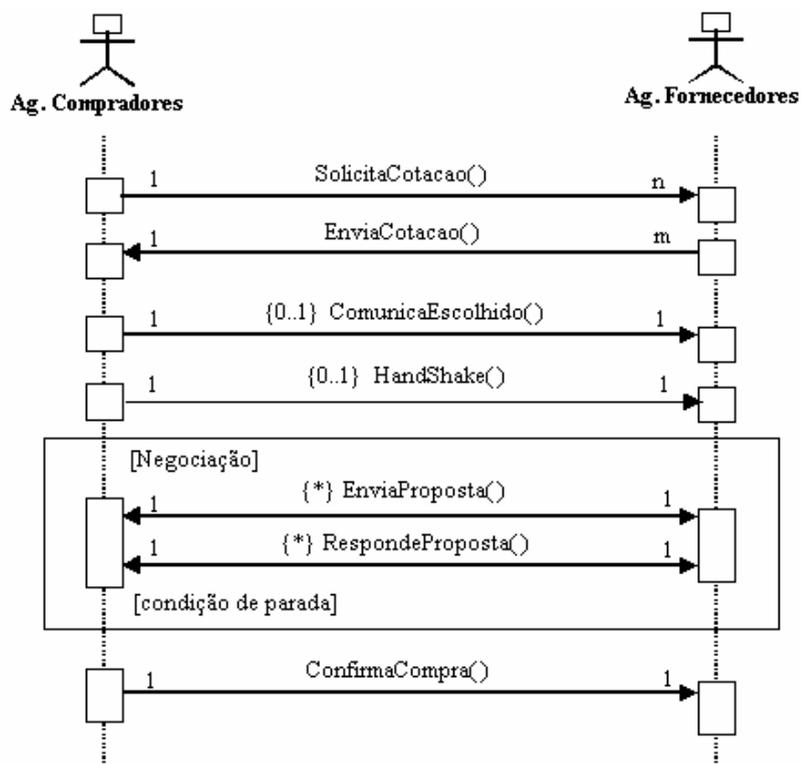


Figura 17 - Diagrama de Seqüência: Ag. Compradores e Ag. Fornecedores

Alguns comentários são necessários para um bom entendimento do diagrama acima. O primeiro deles é sobre cardinalidade. Em todos os outros diagramas, a cardinalidade das relações era da forma um-para-um. Agora, vêem-se também relações do tipo um-para-muitos (ou 1 para n). Por exemplo, quando o Agente Comprador solicita cotações, ele pode estar solicitando para n fornecedores. Dos n fornecedores, m podem enviar cotações para o Agente Comprador ($m \leq n$). Entretanto, observe que a mensagem que indica qual foi o Agente Fornecedor escolhido volta a ter cardinalidade um-para-um, ou seja, o Agente Comprador envia a mensagem “ComunicaEscolhido()” apenas para aquele que ganhou a concorrência.

Um outro ponto interessante são os símbolos $\{0..1\}$ na mensagem “ComunicaEscolhido()”. Isto quer dizer que nenhum ou, no máximo, um Agente Fornecedor vai recebê-la.

O retângulo que envolve as mensagens “EnviaProposta()” e “RespondeProposta()”, juntamente com os rótulos “[Negociação]” e [condição de parada], indicam que há uma estrutura de repetição entre os agentes. O que acontece aqui é que, os negociadores (Comprador e Fornecedor) alternam-se, em sucessivos envios de propostas, por parte de um deles, até o outro aceitar algum dos acordos possíveis contido em uma proposta. Uma vez aceito algum acordo de uma proposta, este passa a ser o acordo atual e os papéis são invertidos: quem propunha, escolhe e vice-versa. Esse processo alternado de envio de proposta e escolha de novo acordo só termina quando a condição de parada do algoritmo implementado for satisfeita.

5.6 DIAGRAMAS DE ATIVIDADES

Há outros diagramas na UML que mostram o comportamento dinâmico dos sistemas modelados. Dois deles são o Diagrama de Estados e o Diagrama de Atividades. Os dois têm muito em comum, em especial, têm estados e transições de estados. A diferença é que no primeiro, o foco está nas transições dos estados, evidenciando os eventos que causam tais transições. Uma outra diferença é que, no diagrama de atividades, os estados são geralmente do tipo “estado de ação”. Um estado é como se fosse uma fotografia de um sistema em um determinado momento. Já um “estado de ação” demonstra que operação está sendo realizada naquele momento do sistema. Para acompanhar tal tipo de diagrama, observam-se os estados de ação e vê-se que a transição de um estado de ação para outro é feita pela finalização do primeiro estado, sem necessidades de eventos externos para haver mudanças de estado. Não

obstante, pode-se utilizar estados “normais” no Diagrama de Atividades, bem como estados de ação no Diagrama de Estados.

Os Diagramas de Atividades mostram o fluxo de operações necessário para se atingir um determinado fim. A circunferência “cheia” indica o início das atividades. A “cheia” dentro de uma “vazia”, indica o término. Assim, escolheu-se o fluxo principal de operações de cada um dos agentes de software, de forma que fique claro como devem ser os seus funcionamentos.

O primeiro diagrama mostra como deve ser o fluxo de operação “normal” de um Agente Comprador:

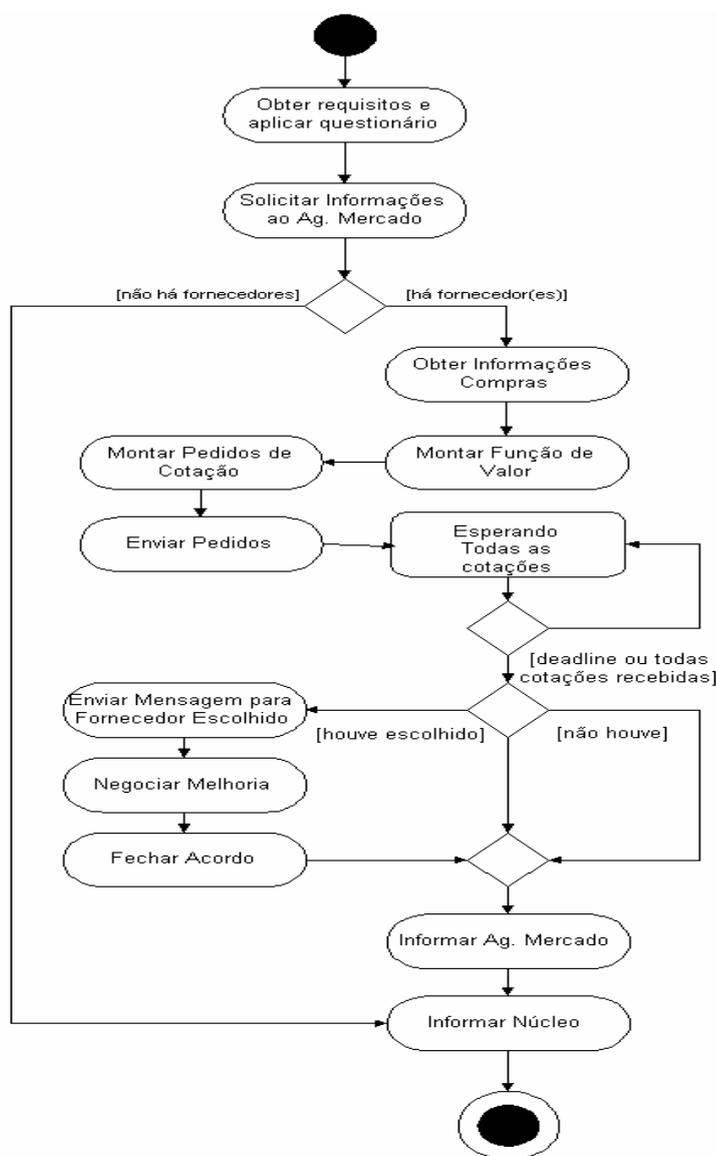


Figura 18 - Diagrama de Atividades de um Ag. Comprador

O próximo diagrama mostra as atividades realizadas por um Agente Fornecedor:

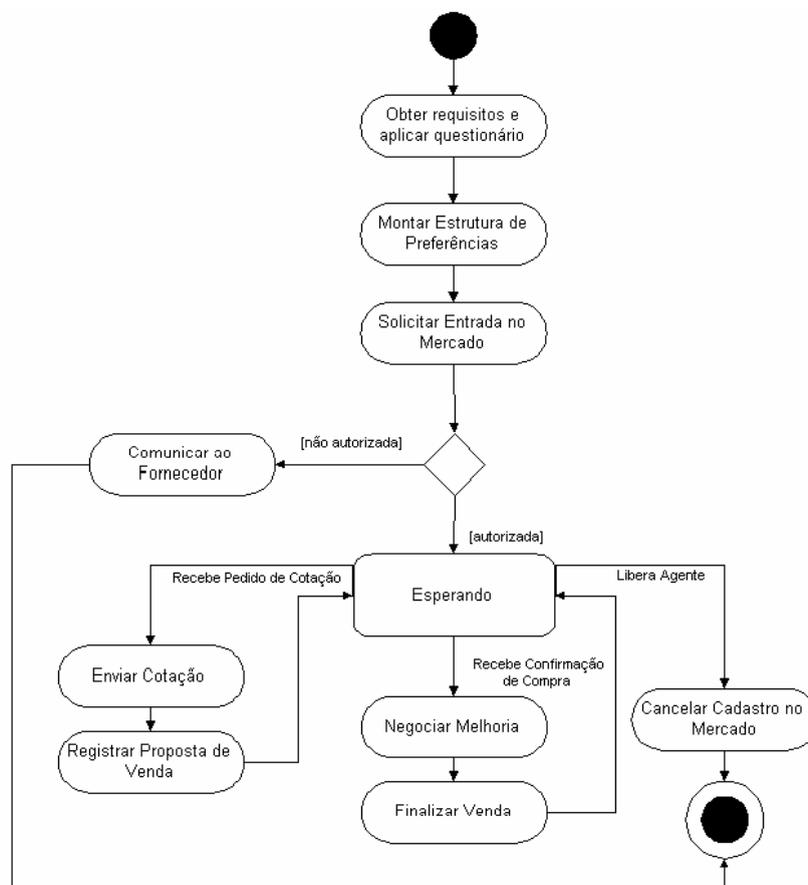


Figura 19 - Diagrama de Atividades de um Ag. Fornecedor

Por fim, apresentam-se as atividades que devem ser desempenhadas pelo Ag. Mercado quanto ao registro de agentes, tanto Ag. Fornecedores como Ag. Compradores. Note-se que não houve a representação do término das atividades. Há outras atividades que não estão sendo representadas na figura.

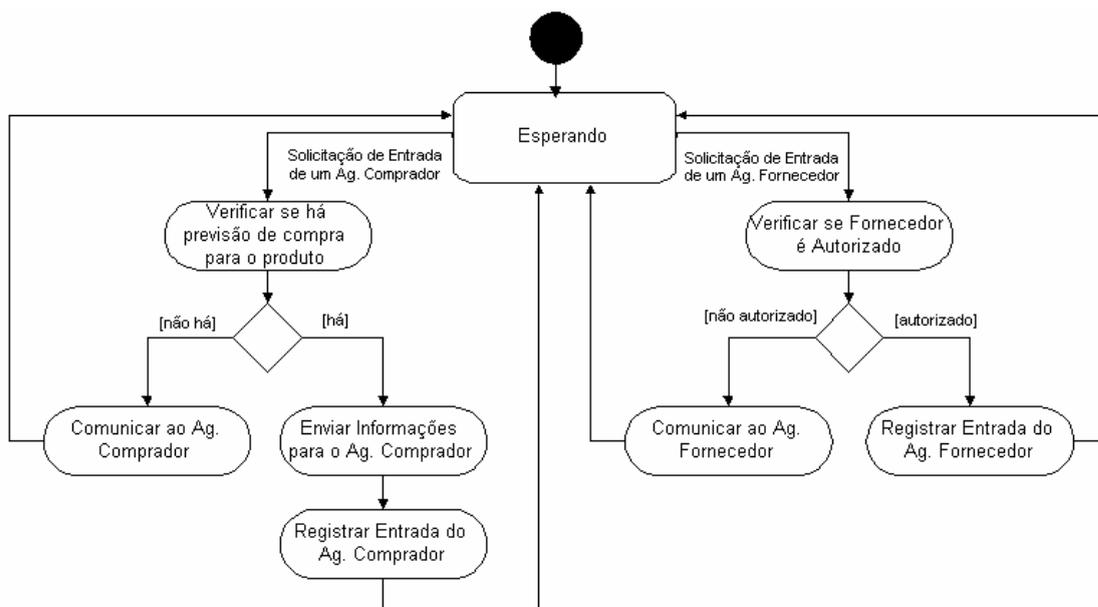


Figura 20 - Diagrama de Atividades do Ag. Mercado (Registrando Agentes)

Outros diagramas quaisquer da UML poderiam ter sido inseridos neste capítulo. Certamente, Diagramas de Classe, por exemplo, serão necessários quando houver uma preocupação maior de como o sistema será implementado. Alguma dúvida sobre o sistema pode implicar necessidade de mais diagramas para, por exemplo, retirar alguma ambigüidade. O entendimento de como deve ser desenvolvido o sistema, entretanto, foi o objetivo desta modelagem, feita através da *Agent UML*.

Os mecanismos de decisão e negociação ainda carecem de maiores explicações, o que é o propósito do próximo capítulo, que, além de propor tais mecanismos, os implementa, analisando seus resultados.

6 MECANISMOS DE DECISÃO E NEGOCIAÇÃO

A decisão de compra obedecerá a dois tipos de critérios: eliminatórios (ou restritivos) e classificatórios (ou de ordenação). O Agente Fornecedor que obedecer a todos os critérios restritivos e obtiver a melhor classificação, será escolhido para fornecer o pedido de compra.

Quando se cria um Agente Comprador, definem-se as faixas de valores que as variáveis de decisão (preço, prazo de entrega e quantidade) podem assumir, bem como, se há restrições sobre as marcas aceitáveis. Assim, por exemplo, determina-se que o valor para a variável prazo pode ser qualquer um compreendido entre 1 e 15 dias. Qualquer cotação que apresentar prazo distinto será eliminada. De forma semelhante, definem-se limites mínimos e máximos para as outras variáveis. Até a variável preço assume um valor mínimo, que é acreditado como sendo o menor valor possível de ser praticado, sem que se suspeite da integridade do Fornecedor.

Obedecidos todos os critérios restritivos, as cotações serão avaliadas através da soma ponderada de todas as funções de valor associadas a cada uma das variáveis de decisão (preço, prazo de entrega, quantidade e qualidade). A cotação que maximizar a função abaixo, que será chamada de Função Valor Total, será escolhida. Em caso de empate, ganha a cotação mais antiga.

$$\text{FuncaoValorTotal} = \sum_{i=1}^4 m_i F_i(x_i)$$

onde:

$i \in \{\text{preço, prazo de pagamento, quantidade, qualidade}\}$

$m_i \rightarrow$ peso relacionado a i

$$\sum_{i=1}^4 m_i = 1$$

$F_i \rightarrow$ tipo de função relacionado a i (linear, exponencial ou inversa)

$x_i \rightarrow$ valor numérico relacionado a i

O conjunto dos pesos será informado na criação do próprio Agente Comprador. É uma etapa de fundamental importância e é realizada através do julgamento do criador do agente.

6.1 CONSTRUINDO AS FUNÇÕES DE VALOR

Para cada variável de decisão de compra, sua função de valor retornará valores entre 0 e 1. O pior valor possível que uma determinada variável possa assumir, retornará 0. O melhor, 1. A escolha desse intervalo é arbitrária, podendo ser qualquer outro intervalo contínuo (com valores pertencentes ao conjunto dos números reais), pois a estrutura de preferência poderia ser reescalada.

Keeney e Raiffa (1976) sugerem um método interativo e iterativo para a construção de uma função de utilidade, bem como uma função de valor. O tomador de decisão deve ser entrevistado para evidenciar alguns pontos da sua função. Inicialmente, procura-se descobrir os valores que devem retornar 0 e 1 (pior e melhor, respectivamente). Depois, questiona-se sobre o valor V , que deixaria o tomador de decisão igualmente satisfeito em sair do pior valor para V ou, de V para o melhor, ou seja, o valor que retorna 0,5. Repete-se o passo anterior para os dois novos intervalos, 0 a 0,5 e 0,5 a 1, e, assim, sucessivamente, até o entrevistador (quem estrutura a decisão) achar que tem pontos suficientes para descrever a função de preferência do decisor.

Poder-se-ia dotar o Agente Comprador com a tarefa de entrevistar o pessoal do Núcleo e/ou da DMS para montar as funções, de forma semelhante ao exposto acima. Conquanto, acredita-se que este processo é lento e pode frustrar o usuário do sistema multiagente.

Uma solução alternativa é predefinir alguns comportamentos. Ao invés de buscar pontos intermediários, pede-se para o decisor escolher, dentre os comportamentos predefinidos, qual o mais próximo da sua “real” função de preferências. Como forma de parametrização, o tomador de decisão pode definir o grau de concavidade da sua curva.

Abordagens semelhantes são facilmente encontradas na literatura, por exemplo, para definir quão rápido um agente negociador cede (CHAVEZ e MAES, 1996; DE PAULA 2000). Em uma abordagem mais geral, Clemen (1996, p.477) mostra como modelar atitudes de riscos (funções de utilidade), através de funções matemáticas (em especial, exponenciais e logarítmicas).

Nesta modelagem, decidiu-se oferecer três tipos de comportamentos para descrever as funções dos tomadores de decisão: linear, exponencial e inversa (do tipo, $y=1/x$). Questionam-se, para o decisor, os limites das funções (ou, em outras palavras, os valores pior e melhor), os tipos das mesmas e, para as exponenciais e inversas, os respectivos graus de concavidade, que representam a rapidez com que as funções associadas crescem ou decrescem, também conhecidos como parâmetros de forma.

A escolha pela função linear foi pela possibilidade do decisor indicar que a taxa crescimento (ou decrescimento) é constante. As exponenciais e inversas foram escolhidas, arbitrariamente, como duas funções que apresentam taxas de crescimento variáveis, tendo as primeiras, retornos sempre menores ou iguais a linear e, as do segundo tipo, sempre maiores ou iguais, para os graus de concavidades admitidos (pertencentes ao conjunto dos números naturais).

É importante salientar que as funções foram escolhidas para simular o uso dos mecanismos de decisão e de negociação. Não houve preocupações com reais representações das funções de valor de qualquer empresa.

Com os limites, têm-se dois pontos: $(x_1, 0)$ e $(x_2, 1)$, onde, x_1 é o pior valor e x_2 é o melhor. Para determinar uma função linear, não se necessita de mais nada. A função tem a seguinte forma, onde x é o valor da variável que se deseja conhecer seu retorno:

$$Linear(x) = \frac{1}{x_1 - x_2} (x_1 - x)$$

Ao contrário da função linear, funções exponenciais e inversas não são definidas com apenas dois pontos, pois várias delas podem passar por estes dois pontos. Para a definição da função exponencial, os passos foram: expressar uma função exponencial “genérica”, substituir os dois pontos na expressão e resolver o sistema. Chamando o pior valor de x_1 , o melhor de x_2 e o grau de concavidade de a , tem-se a seguinte função exponencial:

$$\text{Exponencial}(x) = Ce^{Kx^a} - 1$$

onde:

$$C = \frac{1}{e^{Kx_1^a}}$$

$$K = \frac{\ln(2)}{x_2^a - x_1^a}$$

De forma semelhante à exponencial, a inversa foi determinada, chegando-se a particular função inversa:

$$\text{Inversa}(x) = -\frac{x_1^a x_2^a}{x^a (x_2^a - x_1^a)} + \frac{x_2^a}{x_2^a - x_1^a}$$

Para facilitar a associação da função “real” com uma das predefinidas, sugere-se uma interface que mostre as funções, com a concavidade escolhida. Uma outra sugestão é trazer um texto que “traduza” a função matemática em uma linguagem mais facilmente entendida. Por exemplo, uma função linear para o preço poderia apresentar um texto explicativo do tipo: “quanto mais caro, pior”. Já, ainda para o preço, uma exponencial: “um pouco mais caro, piora um pouco, mas, ao passo que se distancia do melhor valor, vai piorando muito mais rápido”. Para uma inversa, poderia ser: “quanto mais caro, muito pior”.

A Figura 21 traz exemplos das funções exponenciais e inversas (todas crescentes), com diferentes graus de concavidade (2, 4, 8 e 12). A interface gráfica do sistema deve permitir que os criadores dos agentes visualizem os gráficos, podendo manipular o grau de concavidade. Os gráficos das funções lineares (retas) também devem ser mostrados.

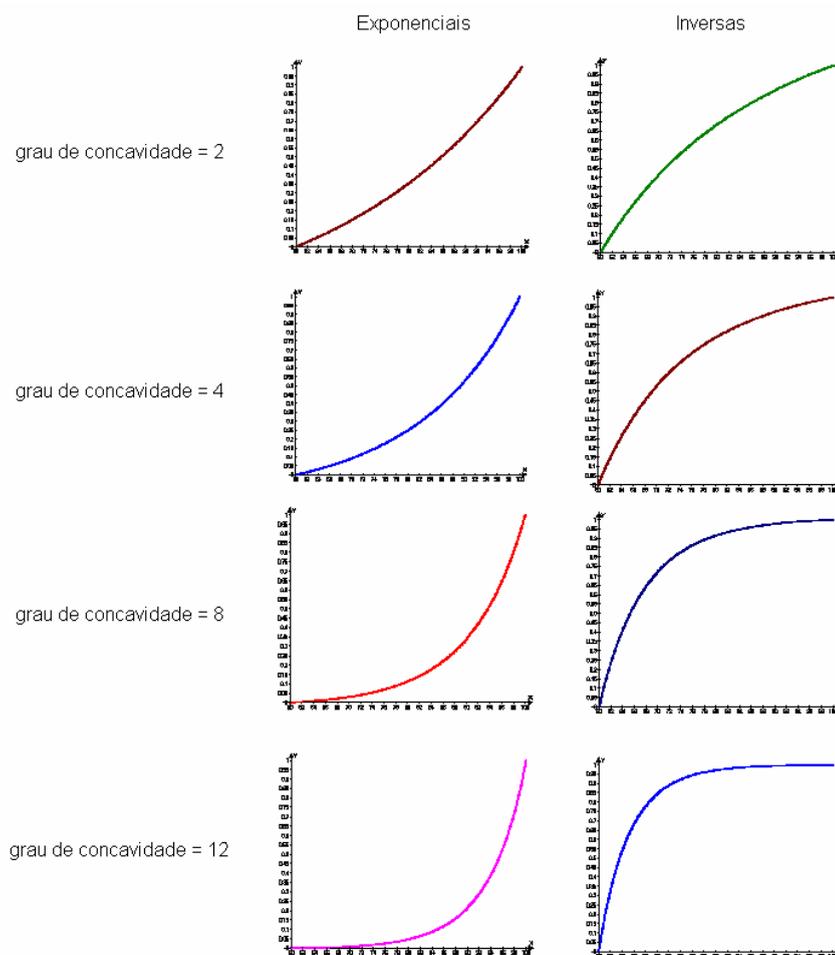


Figura 21 - Exemplos das Funções Exponenciais e Inversas

Assumiu-se que, para o Comprador, as funções para o preço, prazo de entrega e quantidade, são sempre decrescentes. Sendo mais preciso, são sempre não-crescentes, pois podem ser constantes.

O algoritmo para a negociação, apresentado na seção seguinte, exige que ambos os negociadores estejam munidos de funções de valor para expressar suas preferências. Assim, necessita-se construir funções duais às dos compradores para os fornecedores. Por exemplo, se as funções dos compradores são não-crescentes, as funções dos fornecedores serão assumidas como não-decrescentes. A forma de interação sugerida com o usuário é a mesma.

Também será preciso, para o algoritmo de negociação, achar as inversas das funções apresentadas, pois, para este contexto em particular, será necessário encontrar o valor do preço, dada a sua função de valor. Assim, a seguir, apresentam-se as inversas das funções linear, exponencial e inversa, respectivamente:

$$Inv_Linear(x) = x_1 - x(x_1 - x_2)$$

$$Inv_Exponencial(x) = \left[\frac{x_2^a - x_1^a}{\ln(2)} \ln(x+1) + x_1^a \right]^{\frac{1}{a}}$$

$$Inv_Inversa(x) = \left(\frac{C}{x-K} \right)^{\frac{1}{a}}$$

onde:

$$C = -Kx_1^a$$

$$K = \frac{x_2^a}{x_2^a - x_1^a}$$

6.1.1 Os limites das funções dos negociadores

Na seção 6.2, será apresentada a implementação do mecanismo de decisão, que se baseia nas funções de valor dos negociadores. Para haver negociação, é necessário que haja alguma interseção entre os domínios das funções. Isso, contudo, não é suficiente para um adequado funcionamento do mecanismo.

Suponha que o Fornecedor tenha, como limites da sua função de valor associada a variável quantidade, os valores 1 e 300, por exemplo. Por sua vez, o Comprador deseja comprar 100 unidades, tolerando até, por exemplo, 120. Enquanto o intervalo do Fornecedor tem 300 elementos, o do Comprador tem apenas 21. Nesta configuração, caso se esteja avaliando o impacto na Função Valor Total de aumentar de 120 para 121 unidades, ter-se-ia um considerável impacto para o Comprador, enquanto, para o Fornecedor, o impacto seria desprezível, para quase todas os comportamentos possíveis da função de valor deste último.

Para evitar que ocorram estas incompatibilidades no tamanho dos domínios das funções, buscou-se amenizar o problema, através de uma etapa batizada de “handshake”, onde o Comprador informa quais seus limites máximos. Para a quantidade, em particular, o Fornecedor também saberá o limite mínimo, que é a quantidade cotada. Para o prazo, admite-se que o limite inferior para o Comprador é igual a 1 dia. Se o Fornecedor tiver o mesmo prazo mínimo, os intervalos serão os mesmos. Senão, serão, ao menos, compatíveis. No

preço, dificilmente os dois terão os mesmos intervalos, mas também serão intervalos compatíveis, viabilizando possíveis melhorias pelo algoritmo.

6.2 O ALGORITMO PARA NEGOCIAÇÃO INTEGRATIVA

Raiffa (1982) apresentou um método para negociação colaborativa (ou integrativa), consistindo em um negociador apresentar proposições de dois pontos de indiferença ao ponto atual da negociação (ver Capítulo 3): um ponto, um pouco à direita, e um outro, um pouco à esquerda. O outro negociador poderia escolher um dos dois pontos (o de maior retorno de sua função de valor/utilidade) para ser o novo ponto atual da negociação. O processo é repetido, sucessivas vezes, alternando o negociador proponente.

Para que fosse possível a utilização do método, ambos os negociadores necessitavam ter funções de valor (ou de utilidade) como forma de ordenar suas preferências, ou, ao menos, fazer inferência de pontos indiferentes, como se houvesse uma Curva de Indiferença.

No texto citado, não há uma formalização (em notação matemática ou computacional) desse método. A apresentação do mesmo foi feita para duas variáveis. Assim, duas coisas são necessárias para utilizá-lo na negociação integrativa de compras: formalizá-lo (um algoritmo) e expandi-lo para três variáveis (preço, prazo de entrega e quantidade). A variável qualidade não entra na negociação, pois a modelagem prevê que cada Agente Fornecedor venda um produto de uma determinada marca (consequentemente, qualidade). Assim, ela só é usada no processo de decisão: escolher melhor cotação.

Vale lembrar, como visto na modelagem do capítulo anterior, que a negociação é feita apenas entre o comprador e o fornecedor vencedor da concorrência. Será desconsiderada, no mecanismo de negociação proposto, a discussão sobre o impacto da abertura dos valores de reserva de cada negociador.

A seguir, a implementação, em Java, do algoritmo sugerido:

```

Public void negociacao()
{
    Negociador proponente = comprador;
    Negociador decisor = fornecedorEscolhido;
    Negociador aux;
    Proposta propostaAtual;
    Atributos atributosAtual;

    do
    {
        propostaAtual = proponente.enviaProposta();
        if (propostaAtual == null)
        {
            break;
        }
        atributosAtual = decisor.escolheProposta(propostaAtual);
        if (atributosAtual != null)
        {
            decisor.aceitaProposta(atributosAtual);
            proponente.aceitaProposta(atributosAtual);
            aux = decisor;
            decisor = proponente;
            proponente = aux;
        }
    }
    while (parada.satisfeito());
}

```

Figura 22 - Implementação do Algoritmo de Negociação

Na implementação acima, há uma classe chamada *Negociador* que se especializa em *Comprador* e em *Fornecedor*. Para futura implementação em outros contextos, bastam novas especializações desta classe.

A classe *Proposta* é uma lista de objetos da classe *Atributos*, que são, basicamente, conjuntos dos valores das variáveis em negociação (neste caso, preço, prazo de pagamento e quantidade). Esses objetos da classe *Atributos* representam, neste instante, os possíveis acordos que um negociador oferece ao outro: são os acordos indiferentes em preferência. Assim, um objeto da classe *Proposta* é um conjunto de possíveis acordos, enviados pelo *proponente*, para apreciação do *decisor*.

Os acordos indiferentes em preferência são gerados pela manipulação das variáveis discretas (prazo de pagamento e quantidade), somando-se ou subtraindo-se um fator (igual a 1) e, através da variável contínua preço, são feitos os ajustes, de forma a manter-se o mesmo retorno da Função Valor Total. A seguir, a implementação deste método (*enviaProposta*), que ainda precisa ser generalizado para a aplicação em outros contextos, com, por exemplo, n variáveis, discretas e contínuas.

```

public Proposta enviaProposta()
{
    Proposta proposta = new Proposta();

    proposta.inserir(this.adicionaAtributos(fatorIncremento,0));
    proposta.inserir(this.adicionaAtributos(-fatorIncremento,0));
    proposta.inserir(this.adicionaAtributos(0,fatorIncremento));
    proposta.inserir(this.adicionaAtributos(0,-fatorIncremento));
    proposta.inserir(this.adicionaAtributos(fatorIncremento,fatorIncremento));
    proposta.inserir(this.adicionaAtributos(fatorIncremento,-fatorIncremento));
    proposta.inserir(this.adicionaAtributos(-fatorIncremento,fatorIncremento));
    proposta.inserir(this.adicionaAtributos(-fatorIncremento,-fatorIncremento));

    for(int i=1; i< fatorIncremento; i++)
    {
        proposta.inserir(this.adicionaAtributos(fatorIncremento,i));
        proposta.inserir(this.adicionaAtributos(-fatorIncremento,i));
        proposta.inserir(this.adicionaAtributos(fatorIncremento,-i));
        proposta.inserir(this.adicionaAtributos(-fatorIncremento,-i));

        proposta.inserir(this.adicionaAtributos(i,fatorIncremento));
        proposta.inserir(this.adicionaAtributos(i,-fatorIncremento));
        proposta.inserir(this.adicionaAtributos(-i,fatorIncremento));
        proposta.inserir(this.adicionaAtributos(-i,-fatorIncremento));
    }
    this.removeAtributosProposta(proposta);
    this.fatorIncremento++;

    return proposta;
}

```

Figura 23 - Implementação do Método enviaProposta()

Para entender a implementação acima, é necessário entender os métodos *adicionaAtributos()* e *removeAtributosProposta()*. O primeiro, realizado para cada possível acordo, recebe dois parâmetros: incremento na quantidade e incremento no prazo, que podem assumir zero (0), “+*fatorIncremento*” ou “-*fatorIncremento*”. Como houve um aumento, ou uma diminuição, dos valores da variável quantidade, prazo, ou de ambas, mudou-se o retorno da Função Valor Total do negociador. Para que o retorno continue o mesmo, necessita-se de ajuste no preço.

O ajuste no preço é feito através de um método que, em palavras, calcula a diferença entre os retornos das Funções Valor Total do acordo atual e do possível acordo em questão, aplicando o retorno da função preço do acordo atual, somado a esta diferença, na função inversa do preço, achando qual deve ser o novo valor para preço no possível acordo, de tal sorte que a Função Valor Total continue a mesma, ou seja, o possível acordo continue indiferente em preferência ao atual, para o negociador proponente.

O valor de *fatorIncremento* é, inicialmente, igual a 1. Assim, em um primeira “rodada” de oferta de possíveis acordos (ou proposta), um negociador propõe oito possíveis acordos indiferentes em preferência. São os acordos mais próximos do acordo atual, se está se pensando em uma “zona matemática de acordo”.

Caso nenhum dos possíveis acordos seja aceito pelo negociador parceiro (*decisor*), o negociador proponente irá buscar possíveis acordos um pouco mais distantes na “zona matemática de acordo”. Assim, o *fatorIncremento* é adicionado em uma unidade de incremento, que, nos testes, era 1. Essa busca por acordos indiferentes é feita até o negociador *decisor* aceitar algum novo acordo ou, no algoritmo de negociação, ter se chegado a condição de parada (*parada.satisfeito() == true*).

O método *removeAtributosProposta()* simplesmente elimina os possíveis acordos gerados que passaram dos limites das funções, ou seja, verifica se os valores para quantidade, prazo de pagamento e preço, são valores aceitos pelos critérios restritivos. Os possíveis acordos gerados que não satisfaçam os critérios de restrição são eliminados da lista *Proposta*.

Para entender a implementação do algoritmo de negociação, ainda são necessárias explicações sobre os métodos *escolheProposta()* e *aceitaProposta()*. O primeiro, que é realizado pelo negociador *decisor*, calcula todos os possíveis acordos enviados pelo negociador proponente, em busca daquele que maximize sua Função Valor Total, que será o novo acordo. Entretanto, se nenhum dos possíveis acordos propostos aumenta a Função Valor Total do negociador *decisor*, ele espera por uma próxima “rodada” de possíveis acordos oferecidos pelo negociador proponente.

O método *aceitaProposta()* é usado quando, no método anterior, acha-se um acordo melhor, que passa a ser o atual, começando todo o processo novamente, sendo que os negociadores invertem os papéis: vira *decisor* quem era proponente e vice-versa. O *fatorIncremento* retorna a valer 1, como de início.

Na Figura 23, a palavra “*this*” é apenas um detalhe de implementação que serve para distinguir sobre qual negociador se está tratando.

6.3 A IMPLEMENTAÇÃO

As implementações dos mecanismos foram feitas na linguagem de programação Java, em aproximadamente 1700 linhas, independentemente dos conceitos de sistemas multiagentes. Buscou-se, também, torná-las independente em relação à aplicação no contexto de compras. Para adaptá-las em outras aplicações, não são necessários grandes esforços, mas apenas alguns ajustes.

Com a estrutura de classes programada, pode-se, com algumas poucas adaptações, utilizar os mecanismos de decisão e de negociação para a utilização em outros contextos, com “n” variáveis, além de incluir novos comportamentos (funções), como, por exemplo, parabólicos ou logarítmicos.

A interface gráfica não foi implementada. Todos os testes foram feitos no próprio ambiente de programação (Borland JBuilder 6.0), através da interface padrão (caracteres).

6.4 OS RESULTADOS

Escolheram-se quatro produtos que constam no APÊNDICE A. O critério de escolha foi a observação de produtos com características distintas de compra. A tabela encontrada neste apêndice traz a quantidade estimada que será necessário comprar, suficiente para os próximos seis meses. Assim, como se propõe comprar sob demanda, diminuiu-se bastante a quantidade, próximo ao número necessário, hipoteticamente, para um mês. Os valores dados nos testes para a variável preço são compatíveis com os valores expressos na mesma tabela.

Para cada um dos produtos escolhidos, criou-se um cenário para testes. Em cada cenário, escolheram-se três hipotéticos fornecedores candidatos. Os valores cotados para as variáveis foram escolhidos, observando a coerência em relação a possíveis reais fornecedores. As funções, entretanto, foram definidas sem a preocupação de corresponder a realidade de qualquer empresa, servindo para simular a negociação. De forma semelhante, todos os outros “inputs” do sistema obedeceram à mesma sistemática.

Usou-se uma planilha Excel, configurada para, dados os valores das variáveis, seus pesos e tipos de funções associadas, exibir o retorno das Funções Valor Total. Todas as avaliações das cotações feitas pelo programa e pela planilha tiveram o mesmo retorno das funções, para ambos os negociadores. Os valores dos acordos chegados, após a negociação, também foram verificados pela planilha. Todos os valores das Funções Valor Total da implementação e da planilha foram coincidentes.

A seguir, serão apresentados os cenários de testes, através de três tabelas, cada. Na primeira, os “inputs” necessários para configurar o Comprador podem ser observados. Já na segunda, observam-se os “inputs” necessários para configurar os fornecedores. Finalmente, a terceira traz uma comparação entre os acordos, antes e depois da negociação.

6.4.1 Cenário de Testes 1

Para o primeiro cenário, foi escolhido um produto que tivesse variáveis com intervalos relativamente grandes, isto é, o preço, o prazo de entrega e a quantidade têm, potencialmente, boa margem de flexibilidade no momento da negociação. Neste cenário, considerou-se que o prazo e a quantidade tinham importância expressiva na decisão. Coerentemente, o preço continua, todavia, sendo a mais importante das variáveis de decisão. A Tabela 19 mostra as configurações do Comprador.

Tabela 19 - Configuração do Comprador no Cenário de Testes 1

Produto	Cabos para conexões de Hubs 3Com, part number 3C16420				
Quantidade	150				
Marca	Qualidade (0-10)				
-	-				
variável	peso	Função	fator	mínimo	Máximo
Preço	0,60	Inversa	4	80	120
Prazo	0,20	Exponencial	6	1	15
quantidade	0,20	Exponencial	4	150	200
qualidade	-	-	-	-	-

A Tabela 20 apresenta a configuração dos três fornecedores, bem como a avaliação feita pelo Comprador das cotações enviadas pelos mesmos (última coluna). O número do Fornecedor eleito está destacado em negrito, bem como a avaliação da sua cotação, feita pelo Comprador. Somente com este fornecedor que o Comprador realizará a negociação.

Tabela 20 - Configuração e Avaliação dos Fornecedores no Cenário de Testes 1

fornecedor	variável	peso	função	fator	mínimo	máximo	cotação enviada	função valor total para comprador
1	Preço	0,5	Inversa	8	95	120	100	0,558545
	Prazo	0,2	Linear	-	3	15	3	
	Quantidade	0,3	Inversa	8	150	200	150	
	Marca	-	-	-	-	-	-	
2	Preço	0,5	Inversa	6	92	120	98	0,583959
	Prazo	0,15	Exponencial	4	3	15	5	
	Quantidade	0,35	Inversa	8	150	200	150	
	Marca	-	-	-	-	-	-	
3	Preço	0,7	Inversa	4	102	120	102	0,535240
	Prazo	0,0	Linear	-	1	15	1	
	Quantidade	0,3	Inversa	6	150	200	150	
	Marca	-	-	-	-	-	-	

Assim como o Comprador, os fornecedores dão grande importância ao preço. A quantidade também representa, neste cenário, importante papel, tendo, para os três fornecedores, pesos associados relativamente altos. Note-se que quaisquer pequenas alterações iniciais no preço ou na quantidade geram aumentos significativos nas Funções

Valor Total dos fornecedores, pois os três têm funções inversas com altos graus de concavidade associadas a estas variáveis.

Dadas as configurações acima, a Tabela 21 mostra um comparativo entre os acordos anterior e posterior à aplicação do mecanismo de negociação. É importante lembrar que a função qualidade permanece a mesma, após a negociação, pois a marca continua a mesma, sempre. Em particular, neste cenário de testes, não foram determinadas as possíveis marcas. Assim, a variável qualidade não influencia na decisão.

Tabela 21 - Comparativo Antes/Após Negociação no Cenário de Testes 1

	variável	valores	função valor total do comprador	função valor total do comprador (sem qualidade)	função valor total do fornecedor 2
antes da negociação	Preço	98	0,583959	0,583959	0,199073
	Prazo	5			
	Quantidade	150			
depois da negociação	Preço	92,813953	0,605349	0,605349	0,249478
	Prazo	5			
	Quantidade	166			

Neste teste, o algoritmo de negociação conseguiu melhorar o acordo, um pouco para o Comprador, e, significativamente para o Fornecedor.

Fazendo algumas variações nos “inputs” do cenário de testes, pode-se chegar a resultados diferentes. Por exemplo, se o Fornecedor 1 tivesse enviada o preço como 98, assim como fez o Fornecedor 2, ele seria o escolhido na concorrência, pois sua cotação teria sido avaliada pelo Comprador com, apenas, 0,0004 a mais de retorno da Função Valor Total, em relação ao Fornecedor 2. Contudo, após negociação, ter-se-ia chegado a um acordo cuja Função Valor Total do Comprador seria 0,007579 menor.

Alterações nas funções também, evidentemente, podem causar (e geralmente causam) modificações na escolha do fornecedor e/ou nos resultados da negociação. Testou-se, por exemplo, trocar o grau de concavidade da função preço do Fornecedor 2 de 6 para 4 e, depois, para 2. Chegou-se as acordos próximos, mas diferentes.

6.4.2 Cenário de Testes 2

Nos próximos cenários de testes, buscou-se acompanhar o comportamento do algoritmo para a compra de produtos que oferecem pequenas margens para negociação, ou, em outras palavras, que têm funções com pequenos intervalos de domínio. Como o algoritmo

é fundamentado na cessão e ganho dos valores das variáveis, acredita-se que, nestes casos, as melhorias não serão grande, se houver alguma.

No segundo cenário, o produto escolhido foi um Disco Rígido. Também inspirada no APÊNDICE A, a escolha da quantidade foi determinada como quatro unidades, podendo estender-se até seis. A Tabela 22 traz mais detalhes sobre a configuração do Comprador.

Tabela 22 - Configuração do Comprador no Cenário de Testes 2

Produto	Disco rígido de 36.4GB, SCSI para Desktop				
Quantidade	4				
Marca	Qualidade (0-10)				
A	8				
B	9				
C	8				
D	10				
variável	peso	Função	fator	mínimo	máximo
Preço	0,5	Inversa	8	1000	1500
Prazo	0,1	Exponencial	6	1	15
quantidade	0,2	Exponencial	10	4	6
qualidade	0,2	Peso*(qualidade/10)	-	-	-

Note-se que, neste teste, a qualidade está associada às marcas que podem ser compradas. Considerou-se a variável de decisão qualidade importante, pois a mesma recebeu um percentual expressivo na decisão (20%).

A Tabela 23 apresenta os fornecedores que concorrem à compra. Cada um deles vende Disco Rígido de uma marca diferente do outro.

Tabela 23 - Configuração e Avaliação dos Fornecedores no Cenário de Testes 2

fornecedor	variável	peso	função	fator	mínimo	máximo	cotação enviada	função valor total para comprador
1	Preço	0,6	Exponencial	8	1350	1500	1430	0,477647
	Prazo	0,2	Inversa	4	5	15	10	
	Quantidade	0,2	Exponencial	8	4	6	4	
	Marca	B	-	-	-	-	-	
2	Preço	0,7	Inversa	6	1400	1500	1470	0,503371
	Prazo	0,05	Linear	-	4	15	5	
	Quantidade	0,25	Exponencial	2	4	6	4	
	Marca	D	-	-	-	-	-	
3	Preço	0,4	Inversa	4	1300	1500	1350	0,486670
	Prazo	0,2	Exponencial	12	1	15	5	
	Quantidade	0,4	Inversa	6	4	6	4	
	Marca	C	-	-	-	-	-	

Finalmente, a Tabela 24 expõe os dados comparativos entre os acordos anterior e posterior à negociação.

Tabela 24 - Comparativo Antes/Após Negociação no Cenário de Testes 2

	variável	valores	função valor total do comprador	função valor total do comprador (sem qualidade)	Função valor total do fornecedor 2
antes da negociação	Preço	1470	0,503371	0,303371	0,528630
	Prazo	5			
	Quantidade	4			
depois da negociação	Preço	1471,083401	0,503371	0,303371	0,530881
	Prazo	4			
	Quantidade	4			

Como esperado, não houve grandes mudança entre os acordos. A mudança foi quase imperceptível, ocorrendo apenas uma discreta melhoria para o Fornecedor. Já o Comprador, manteve seu resultado da Função Valor Total. O que ocorreu aqui foi a aceitação, pelo Fornecedor, de um dos possíveis acordos oferecidos pelo Comprador, que, por sua vez, não conseguiu oferecer nenhuma proposta nova de acordo que fosse do interesse do Comprador. Observe-se que, além do pequeno intervalo possível da função da variável quantidade, havia também um pequeno intervalo para o prazo. Neste cenário, o Comprador também dificultou a negociação quando definiu suas funções com altos graus de concavidade.

Algumas variações foram feitas neste cenário, mas todas continuaram a indicar que a negociação não melhorava o acordo para o Comprador.

6.4.3 Cenário de Testes 3

No cenário de testes 3, buscou-se analisar a compra de um produto com preços pequenos. A variável de decisão preço tem um peso associado muito alto e um intervalo pequeno para negociação. Não há preocupação com a marca, consequentemente, com a qualidade. A quantidade apresenta um bom intervalo para negociação.

Tabela 25 - Configuração do Comprador no Cenário de Testes 3

Produto	Abraçadeiras de Nylon				
Quantidade	100				
Marca	Qualidade (0-10)				
-	-				
Variável	peso	Função	fator	mínimo	máximo
Preço	0,8	Exponencial	8	0,10	0,17
Prazo	0,1	Linear	-	1	15
Quantidade	0,1	Exponencial	4	100	120
Qualidade	0	-	-	-	-

A Tabela 26 apresenta quais as configurações para os fornecedores.

Tabela 26 - Configuração e Avaliação dos Fornecedores no Cenário de Testes 3

Fornecedor	variável	peso	função	fator	mínimo	máximo	Cotação enviada	função valor total para comprador
1	Preço	0,6	Exponencial	8	0,14	0,17	0,15	0,648220
	Prazo	0	Linear	-	1	15	1	
	Quantidade	0,4	Exponencial	8	100	120	100	
	Marca	-	-	-	-	-	-	
2	Preço	0,7	Inversa	6	0,13	0,17	0,14	0,764220
	Prazo	0,15	Linear	-	1	15	5	
	Quantidade	0,15	Inversa	8	100	120	100	
	Marca	-	-	-	-	-	-	
3	Preço	0,7	Inversa	4	0,16	0,17	0,18	-
	Prazo	0,05	Exponencial	12	1	15	2	
	Quantidade	0,25	Inversa	6	100	120	100	
	Marca	-	-	-	-	-	-	

O Fornecedor 3 não obedeceu aos critérios restritivos. Assim, foi eliminado da concorrência, por isso sua Função Valor Total não aparece na tabela. Observe-se também que o Fornecedor escolhido tem, assim como o Comprador, um peso alto associado à variável preço (70% de importância total da compra). O resultado do processo de negociação é apresentado na Tabela 27.

Tabela 27 - Comparativo Antes/Após Negociação no Cenário de Testes 3

	Variável	valores	função valor total do comprador	função valor total do comprador (sem qualidade)	função valor total do fornecedor 2
antes da negociação	Preço	0,14	0,764220	0,764220	0,356927
	Prazo	5			
	Quantidade	100			
depois da negociação	Preço	0,140173	0,769966	0,769966	0,370842
	Prazo	1			
	Quantidade	104			

Houve um aumento ínfimo na Função Valor Total do Comprador e, um pequeno na do Fornecedor. Mesmo com pequenas possibilidades de melhorias, o processo de negociação fez um ajuste mínimo nas variáveis. Note-se que, na prática, não houve mudança na variável preço.

Além dos pequenos intervalos das funções, o que dificultou a melhoria do acordo pelo algoritmo de negociação foi o alto valor dado ao peso da variável preço. Quando se testou o cenário, com alterações nos pesos do comprador, sendo 0,6 para o preço e 0,2 para prazo e quantidade, houve melhorias significativas nos acordos antes e depois da negociação.

6.4.4 Teste 4

Seguindo a linha de testar os mecanismos de decisão e negociação para produtos com funções de valor com pequenos intervalos associados, o quarto cenário de testes apresentou

uma situação curiosa: um expressivo aumento da Função Valor Total do Fornecedor, mediante nenhum aumento da função do Comprador. Seguem as informações, na Tabela 28, sobre o Comprador.

Tabela 28 - Configuração do Comprador no Cenário de Testes 4

Produto	No-break de 3,5 KVA				
Quantidade	1				
Marca	Qualidade (0-10)				
A	10				
B	9,5				
C	8				
variável	peso	Função	Fator	mínimo	máximo
Preço	0,5	Inversa	4	4500	5800
Prazo	0,05	Exponencial	6	1	30
quantidade	0,15	Exponencial	4	1	3
qualidade	0,3	Peso*(qualidade/10)	-	-	-

Na Tabela 29, as informações sobre os fornecedores.

Tabela 29 - Configuração e Avaliação dos Fornecedores no Cenário de Testes 4

fornecedor	variável	peso	Função	fator	mínimo	máximo	Cotação enviada	função valor total para comprador
1	Preço	0,7	Exponencial	8	5350	5800	5500	0,567253
	Prazo	0	Linear	-	1	30	1	
	Quantidade	0,3	Exponencial	8	1	3	1	
	Marca	A	-	-	-	-	-	
2	Preço	0,7	Inversa	6	5100	5800	5200	0,640634
	Prazo	0,15	Exponencial	4	3	30	5	
	Quantidade	0,15	Inversa	8	1	3	1	
	Marca	B	-	-	-	-	-	
3	Preço	0,6	Inversa	4	4900	5800	5000	0,670323
	Prazo	0,15	Exponencial	12	5	30	7	
	Quantidade	0,25	Inversa	6	1	3	1	
	Marca	C	-	-	-	-	-	

Analisando os “inputs” do sistema, pode-se ver que o Comprador pode aumentar sua Função Valor Total das seguintes formas: diminuição do preço ou do prazo (o atual é de 7 dias, podendo melhorar para 6 ou 5). Com pouca margem no prazo, além do peso associado baixo (5%), na prática, a diminuição do preço que poderia resultar em boas melhorias para o Comprador.

O Fornecedor 3, por sua vez, pode aumentar sua Função Valor Total quando aumenta o valor do preço, o prazo ou a quantidade. Uma forma comum em melhorar o acordo para ambos os negociadores seria a diminuição do preço, em troca do aumento do prazo ou da quantidade. Entretanto, esta “manobra” é dificultada pela proximidade do preço cotado pelo Fornecedor 3 (5000) ao seu valor mínimo de venda (4900).

Os resultados da negociação estão na Tabela 30.

Tabela 30 - Comparativo Antes/Após Negociação no Cenário de Testes 4

	variável	valores	função valor total do comprador	função valor total do comprador (sem qualidade)	função valor total o fornecedor 2
antes da negociação	Preço	5000	0,670323	0,430323	0,094946
	Prazo	7			
	Quantidade	1			
depois da negociação	Preço	4914,928015	0,670326	0,430326	0,261223
	Prazo	5			
	Quantidade	2			

Quando se definiu para o Fornecedor 3 que o mínimo valor para venda era igual a 4500, houve uma boa negociação, com ganhos expressivos para ambos os negociadores.

6.5 DISCUSSÃO DOS MECANISMOS

Em todos os testes, o algoritmo de negociação cumpriu seu papel: buscar possíveis melhorias mútuas. Em nenhum dos quatro cenários de testes realizados, houve uma diminuição da Função Valor Total de qualquer negociador. Essa característica é implícita ao fundamento do algoritmo e da negociação colaborativa, de uma forma mais geral. Portanto, a solução apresentada é uma representante das possíveis formas de negociação colaborativa. Também se pôde constatar que a idéia de Raiffa (1982) é passível de implementação para negociação de compras.

Pode-se observar que os resultados da negociação são extremamente sensíveis a todos os “inputs” de configuração. Assim, recomenda-se, em uma aplicação real, um extremo cuidado nas definições de todos estes “inputs”. Isto inclui a definição das funções que, em uma situação real, deve representar bem as estruturas de preferência dos tomadores de decisão. As funções aqui apresentadas e testadas serviram para demonstrar os mecanismos de decisão e negociação propostos, não sendo fundamentadas na realidade de nenhuma empresa.

Uma última observação é que, como teste, na implementação da negociação, inverteu-se a ordem dos negociadores, ficando o Fornecedor responsável pela primeira proposta. Não houve diferenças significativas.

7 CONSIDERAÇÕES FINAIS

Assim como mostrado particularmente para a área de Compras, agentes de software podem ser tecnologias de extrema utilidade nas empresas. O fato de haver delegação de decisão para softwares abre inúmeras possibilidades. Tarefas que tinham necessariamente que ser feitas por pessoas podem ser realizadas por software, se bem definidas e implementadas.

Através de um estudo de caso, pôde-se defrontar com algumas etapas do desenvolvimento de um sistema multiagente para Compras. Duas interessantes discussões paralelas foram levantadas: modelagem pela *Agent UML* e a aplicação de mecanismos de decisão e negociação.

7.1 RESULTADOS ALCANÇADOS

7.1.1 Modelagem

- ◆ A modelagem de um sistema multiagente para a aplicação proposta foi apresentada. Além de representar um grande passo para a construção do sistema em questão, a modelagem serviu para levantar questões sobre a *Agent UML*;
- ◆ A *Agent UML* ainda é um conjunto de propostas para extensão da UML. Assim, ainda há diversas formas diferentes para expressar conceitos idênticos. Evidentemente, isto é o caminho normal para o estabelecimento de um padrão, que é o que se busca;
- ◆ Embora existam diversos esforços, a *Agent UML* não parece ser uma solução ideal, ou próxima disso, para modelar agentes, pois ainda está bem enraizada nos conceitos de objetos, não de agentes. Em casos que não há mecanismos de aprendizado nos agentes, como o proposto neste trabalho, ela ainda parece oferecer bons resultados. Entretanto, mesmo os agentes sem tais mecanismos têm algumas características diferentes de objetos. Assim, uma extensão de uma linguagem para modelar objetos pode não ser razoável para modelar agentes;

7.1.2 Mecanismos de Decisão e Negociação

- ◆ A decisão de qual a melhor cotação recebida para o fornecimento dos produtos desejados foi estruturada através de uma soma ponderada de funções de valor. Para cada variável de decisão de compra, existiam um peso e uma função de valor associados;
- ◆ A negociação apresentou-se, neste contexto, como sendo uma excelente etapa para ajustes dos valores das variáveis que, nitidamente, poderiam ser melhorados. Nos testes, houve maiores melhorias para os fornecedores que para os compradores. Isto é explicado pelas diferenças nos domínios das funções dos negociadores. Os fornecedores, em geral, tiveram funções com domínios bem menores. Assim, um pequeno deslocamento no valor de uma variável representava um maior retorno da função associada para o Fornecedor, em relação ao Comprador;
- ◆ Mesmo sendo o Comprador o foco do sistema proposto, boas melhorias para o Fornecedor ajudam a manter bons relacionamentos com este último, o que também traz vantagens para o primeiro.

7.1.3 Comentários Finais

- ◆ Acredita-se que os objetivos da dissertação foram atingidos, pois houve, no Capítulo 2, uma revisão sobre o uso de agentes de software nas empresas, despertando para os atuais e possíveis usos destas tecnologias. Através de um estudo de caso, pôde-se avaliar a proposta de extensão da UML, *Agent UML*, para modelar sistemas multiagentes. Também foi possível implementar e avaliar as idéias da negociação colaborativa de Raiffa (1982) para o contexto de compra de suprimentos de TI. As discussões sobre as funções de valor e sobre o algoritmo de negociação extrapolaram o contexto proposto e deixaram espaço para futuros trabalhos.

7.2 LIMITAÇÕES DO ESTUDO

7.2.1 Estudo de Caso

- ◆ Toda a modelagem e as implementações dos mecanismos de decisão e negociação foram idealizadas como uma solução alternativa para a compra de suprimentos de TI da Brasil Telecom S.A. Evidentemente, muito pode ser aproveitado para outros contextos.

7.2.2 Modelagem

- ◆ Como não se determinou a forma em que o sistema multiagente deveria ser implementado, não se chegou, na modelagem feita, a níveis mais próximos da implementação. Nem os Diagramas de Classes, por exemplo, foram utilizados, pois já induzem, em demasia, uma implementação orientada a objetos. A forma como o sistema pode vir a ser implementado não faz parte do escopo deste trabalho.

7.2.3 Mecanismos de Decisão e Negociação

- ◆ A escolha das funções exponenciais, lineares e inversas, não foi feita de forma que retratasse a realidade da compras de suprimentos de TI da Brasil Telecom ou de outro processo de compras qualquer;
- ◆ As implementações dos mecanismos foram feitas para o contexto apresentado na modelagem, bem como os cenários de testes foram inspirados nos produtos do APÊNDICE A;
- ◆ A fase de “handshake” foi de fundamental importância para o funcionamento adequado do mecanismo de negociação. Sem esta fase, o Fornecedor provavelmente estabeleceria domínios para suas funções bem maiores que faz o Comprador, inviabilizando a negociação. Todavia, quando o Comprador informa os limites máximos de suas funções, ele abre informações que, *a priori*, não deveriam ser informadas, pois podem levar ao uso malicioso das mesmas por parte dos fornecedores, como elevar, em conjunto, o preço dos produtos para próximo do valor máximo que o Comprador está disposto a pagar. Questões sobre a segurança das informações estão fora do escopo do trabalho.

7.3 RECOMENDAÇÕES PARA ESTUDOS FUTUROS

- ◆ Como continuação da modelagem aqui apresentada, propõe-se apresentar modelos mais próximos da implementação. Para tanto, é necessário que se defina como os agentes serão implementados. Se for através de objetos, a *Agent UML* pode continuar sendo usada. Caso seja decidida outra forma, como uma linguagem própria para agentes, pode haver dificuldade em fazer o elo entre o já feito e os próximos modelos;
- ◆ Um outro passo adiante seria a construção de um protótipo do sistema multiagente modelado;

- ◆ Também como futuro trabalho, pode-se expandir o sistema com outros tipos de agentes, como, por exemplo, um agente para fazer o acompanhamento da entrega do produto;
- ◆ Variações foram feitas nas configurações dos compradores e dos fornecedores. Em alguns casos, como era esperado, a decisão muda. Mecanismos de decisão com análise de sensibilidade poderiam ser pensados em futuros trabalhos;
- ◆ Não se levaram em consideração, neste trabalho, aspectos de segurança da informação. Um deles, de particular interesse neste contexto, é deixar que seu parceiro de negociação descubra suas estruturas de preferência e seus “valores de reserva”. A forma possível para haver a negociação sem a abertura de tais valores é ter todos os agentes pertencentes a uma terceira parte, confiável a ambos os parceiros. Uma nova modelagem, seguindo esta abordagem, pode ser feita.
- ◆ Em uma futura implementação da solução aqui proposta, aconselha-se criar funções sob medida para a empresa, o que pode ser feito de várias maneiras, como, por exemplo, através de entrevistas para descobrir algumas preferências e tentar associar alguma função matemática de acordo com o levantado;
- ◆ Os mecanismos de decisão e negociação propostos nesta dissertação podem ser adaptados facilmente para outros contextos. Como sugestão de futuros trabalhos, têm-se: generalização das classes e métodos utilizados que ainda não são genéricas, estudo aprofundado da complexidade computacional da solução, melhoria dos mecanismos (por exemplo, uso de heurísticas para a determinação das condições de parada ou técnicas de inteligência artificial para aprendizado de quais os melhores caminhos a seguir na “zona de acordo”);
- ◆ Outro desafiador futuro trabalho é generalizar a negociação para vários negociadores. Por exemplo, o Comprador poderia negociar com vários fornecedores, bem como estes últimos, com vários compradores.

8 REFERÊNCIAS BIBLIOGRÁFICAS

ABU-HAKIMA, S., MCFARLAND, C., MEECH, J.F. **An Agent-Based System for Email Highlighting**. Proceedings of the Fifth International Conference on Autonomous Agents (ACM AGENTS'01), Montreal, Quebec, Canadá, p.224-225, 2001.

ALBUS, J., McCAIN, H., LUMIA, R. **NASA/NBS Standard Reference Model for Telerobot Control System Architecture (NASREM)**, NBS Technical Note 1235, Robot System Division, NIST, 1987 apud.

BAILY, P., FARMER, D., JESSOP, D., JONES, D., **Compras – Princípios e Administração**, São Paulo: Atlas, 2000.

BAUER, B., MÜLLER, J.P., ODELL, J. **Agent UML: A Formalism for Specifying Multiagent Interaction**, em **Agent-Oriented Software Engineering**, CIANCARINI, P., WOOLDRIDGE, M. eds., Berlin, Alemanha, p.91-103, 2001.

BECKER, J.L. **Teoria axiomática da utilidade esperada**. Cadernos de Matemática e Estatística, Série C, Nº 11, Porto Alegre, UFRGS, novembro 1988.

BERGENTI, F., POGGI, A. **Exploiting UML in the Design of Multi-Agent Systems**. Proceedings of ESAW Workshop at ECAI, Paris, França, p.106-113 2000.

BRESCIANI, P., PERINI, A., GIORGINI, P. et al. **A Knowledge Level Software Engineering Methodology for Agent Oriented Programming**. Proceedings of the Fifth International Conference on Autonomous Agents (ACM AGENTS'01), Montreal, Quebec, Canadá, p.648-655, 2001.

BUER, B. **UML Class Diagrams Revisited in the Context of Agent-Based Systems**. Proceedings of Agent-Oriented Software Engineering (AOSE 2001), Montreal, Estados Unidos, 2001.

CHAVEZ, A.; MAES, P. **Kasbah: An Agent Marketplace for Buying and Selling Goods**. First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology, London, 1996.

CHEN, Y., PENG, Y., FININ, T. et al. **A negotiation-based Multi-agent System for Supply Chain**, Proceedings of Agents 99 Workshop on Agent Based Decision-Supported for Managing the Internet-Enabled Supply-Chain, Seattle, Estados Unidos, p.15-20, 1999.

CLEMEN, R.T. **Making hard decisions: an introduction to decision analysis**. 2nd ed. Duxbury Press, Belmont, 1996.

CONWAY, D.G., KOEHLER, G.J. **Interface agents: caveat mercator in electronic commerce**. Decision Support Systems Vol. 27, p.355–366, 2000.

COSENTINO, M., CHELLA, A., LO FASO, U. **Design agent-based systems with UML**. International Symposium on Robotics e Automation, México, 2000.

DE PAULA, G.E. **Modelo de Negociação Bilateral para Comércio Eletrônico**. Dissertação de Mestrado, Curso de Pós-Graduação em Ciência da Computação, UFPE, 2000.

DECKER, K., ZHENG, X., SCHMIDT, C. **A Multi-Agent System for Automated Genomic Annotation**. Proceedings of the Fifth International Conference on Autonomous Agents (ACM AGENTS'01), Montreal, Quebec, Canadá, p.433-440, 2001.

DEGRAEVE, Z.; LABRO, E., ROODHOOFT, F. **An evaluation of vendor selection models from a total cost of ownership perspective**. European Journal of Operational Research. Vol.125, p.4-58, 2000.

DUTTA, P.S., DEBNATH, S., SEN, S. **A Shopper's Assistant**. Proceedings of the Fifth International Conference on Autonomous Agents (ACM AGENTS'01), Montreal, Quebec, Canadá, p.59-60, 2001.

FARACO, R.A. **Uma Arquitetura de Agentes para Negociação Dentro do Domínio do Comércio Eletrônico**. Dissertação de Mestrado, Curso de Pós-Graduação em Ciência da Computação, UFSC, 1998.

FLAKE, S., GEIGER, C., KÜSTER, J.M. **Towards UML-based Analysis and Design of Multi-Agent Systems**. International Symposium on Information Science Innovations in Engineering of Natural and Artificial Systems (ENAIIS 2001), Dubai, 2001.

FRANKLIN, S.; GRAESSER, A. **Is it an Agent, or just a Program? A taxonomy for Autonomous Agents**. Proceedings of the Third International Workshop on Agents Theories, Architectures, and Languages, Spring, Verlag, 1996.

GEBAUER, J.; BEAM, C., SEGEV, A. **The Use of Emerging Technologies in Procurement – State of the Art and a Look into the Future**. 9th Int'l Conference of the Information Resources Management Association (IRMA), 1998.

GEBAUER, J.; SEGEV, A. **Emerging Technologies to Support Indirect Procurement**. Information Technology and Management. Vol.1, p.107-128, 2000.

GENESERETH, M.R. **Software Agents**. Communications of the ACM, Vol.37, No.7, July, 1994.

GIESE, L.F. **Estrutura de agentes para os processos de compra e venda utilizando tomada de decisão difusa**. Dissertação de Mestrado, Curso de Pós-Graduação em Ciência da Computação, UFSC, 1998.

GILBERT, D. **Intelligent Agents: The Right Information at the Right Time**. IBM white paper. <http://www.networking.ibm.com/iag/iagwpl.html>. Maio, 1997 apud

GUTTMAN, R.H.; MAES, P. **Cooperative vs. Competitive Multi-Agent Negotiations in Retail Electronic Commerce**. Second International Workshop on Cooperative Information Agents (CIA'98), Paris, 1998.

HERNANDEZ, K. , SEN, S. **A Buyer's Agent**. Proceedings of the Fourth International Conference on Autonomous Agents (ACM AGENT'00), Barcelona, Espanha, p.152-162, 2000.

HERSTEIN, I.N., MILNOR, J. **An axiomatic approach to measurable utility**. Econometrica, Vol. 50, p.277-297, 1953 apud.

HOFFMAN, D.L., NOVAK, T.P. **Marketing in hyper- media computer-mediated environments: conceptual foundations.** Journal of Marketing 60 (2), p.50-68, 1996 apud.

JAIN, B.A., SOLOMON, J.T., **The effect of task complexity and conflict handling styles on computer-supported negotiations.** Information and Management, Vol.37, p.161-168, 2000.

JO, C. **A Seamless Approach to the Agent Development.** Proceedings of 15th Annual Symposium on Applied Computing, Las Vegas, Estados Unidos, p.641-647, 2001.

KARACAPILIDIS, N., MORAÏTIS, P. **Intelligent Agents for an Artificial Market System.** Proceedings of the Fifth International Conference on Autonomous Agents (ACM AGENTS'01), Montreal, Quebec, Canadá, pp. 592-599, 2001.

KEENEY, R.L.; RAIFFA, H. **Decisions with Multiple Objectives: preferences and value trade-offs.** John Wiley & Sons, New York, 1976.

KERN, E. **Uma Estrutura de Agentes para o Processo de Licitação.** Dissertação de Mestrado, Curso de Pós-Graduação em Ciência da Computação, UFSC, 1998.

KERSTEN, G., MICHALOWISK, W., SZPAKOWICZ, S., KOPERCZAK, Z. **Restructurable representations of negotiation,** Management Science 37 (10), p.1269-1290, 1991 apud.

LEWICKI, R., SAUNDERS, D., MINTON, J. **Essentials of Negotiation.** Irwin, 1997 apud.

LIANG, T.P.; HUANG, J.S. **A framework for applying intelligent agents to support electronic trading.** Decision Support Systems, Vol.28, p.305-317, 2000.

LIND, J.. **Specifying Agent Interaction Protocols with Standard UML** Proceedings of the Second International Workshop on Agent-Oriented Software Engineering (AOSE-2001). Heidelberg, Alemanha, 2002.

ODELL, J., PARUNAK, H., BAUER, B., **Representing Agent Interaction Protocols in UML,** em **Agent-Oriented Software Engineering,** CIANCARINI, P., WOOLDRIDGE, M. eds., Berlin, Alemanha, p.121-140, 2001.

PATTIE, M., GUTTMAN, R.H., MOUKAS, A.G., **Agents that Buy and Sell: Transforming Commerce as we Know It**, Communications of the ACM, Vol.43, March 1999.

PREIST, C., BYDE, A., BARTOLINI, C. **Economic Dynamics of Agents in Multiple Auctions**. Proceedings of the Fifth International Conference on Autonomous Agents (ACM AGENTS'01), Montreal, Quebec, Canadá, p.545-551, 2001.

RAIFFA, H. **The Art & Science of Negotiation**. Harvard University Press, 1982.

RAMSEY, F. **The Foundations of Mathematics and Other Logical Essays**. R.B. BRAITHWAITE (ed.). Routledge & Kegan Paul Ltd., London, 1931 apud.

ROSENSCHEIN, J., ZLOTKIN, G. **Rules of Encounter: Designing Conventions for Automated Negotiation among Computers**. MIT Press, 1994 apud.

RUSSEL, J., NORVIG, P. **Artificial Intelligence: A Modern Approach** Prentice Hall, New Jersey, 1995 apud.

SLACK, N. et al. **Administração da Produção**. Editora Atlas S.A., São Paulo, 1997.

SOO, V.W. **Agent Negotiation in Trusted Third Part Mediated Uncertain Games**. Proceedings of the Fourth International Conference on Autonomous Agents (ACM AGENT'00), Barcelona, Espanha, p.265-266, 2000.

STONE, P., VELOSO, M. **Multiagent Systems: A Survey from a Machine Learning Perspective**. *CMU CS technical report number CMU-CS-97-193, USA*, 1997 apud.

TURBAN, E.; LEE, J.K.; KING, D., CHUNG, M. **Electronic Commerce: A Managerial Perspective**. 1st ed. Prentice Hall, New Jersey, 2000.

UML 1.4 Specification, disponível em <http://www.omg.org/technology/documents/formal/uml.htm>, visitado em junho/2002.

von NEUMANN, J.; MORGENSTERN, O. **Theory of games and economic behavior**. New York, Princeton University Press, 1947.

WOOLDRIDGE, M., JENNINGS, N.R., KINNY, D. **A Methodology for Agent-Oriented Analysis and Design.** Proceedings of The Third International Conference on Autonomous Agents (ACM AGENTS'99). Seattle, Estados Unidos, P.69-76, 1999.

WOOLDRIDGE, M., JENNINGS, N.R., KINNY. **The Gaia Methodology for Agent-Oriented Analysis and Design.** Journal of Autonomous Agents and Multi-Agent Systems. 3(3), p.285-312, 2000.

APÊNDICE A – Possível tabela para formatação de um contrato aberto¹

descrição	quantidade	unidade	preço unitário	quant. * preço
Cabo par trançado 24 AWG UTP, categoria 5E	2500	MT	R\$0,45	R\$1.125,00
Disco rígido de 18.2GB, SCSI para Desktop	22	U	R\$745,35	R\$16.397,70
Disco rígido de 20GB, IDE para Desktop	18	U	R\$353,24	R\$6.358,32
Disco rígido de 30GB, IDE para Desktop	45	U	R\$360,85	R\$16.238,25
Disco rígido de 36.4GB, SCSI para Desktop	60	U	R\$1.496,92	R\$89.815,20
Disco rígido de 40GB, IDE para Desktop	55	U	R\$371,65	R\$20.440,75
Disco rígido de 9.1GB, SCSI para Desktop	14	U	R\$620,34	R\$8.684,76
Fonte de alimentação para computadores AT	35	U	R\$30,65	R\$1.072,75
Fonte de alimentação para computadores ATX genérica	40	U	R\$47,25	R\$1.890,00
Cooler AMD MMX Celeron	60	U	R\$12,05	R\$723,00
Cooler ATLON e Pentium III SECC SLOT 1	40	U	R\$13,64	R\$545,60
Cooler para Pentium FC-PGA	55	U	R\$14,21	R\$781,55
Estabilizador de energia de 1 e 2 KVA (entrada bivolt e saída 110v)	35	U	R\$45,25	R\$1.583,75
No-break de 1,2 KVA, frequência 60 Hz	8	U	R\$410,65	R\$3.285,20
No-break de 1,5 KVA	12	U	R\$2.100,69	R\$25.208,28
No-break de 2,5 KVA, com entrada e saída 110V	10	U	R\$3.956,84	R\$39.568,40
No-breaks de 5 KVA, com entrada e saída 110V	10	U	R\$3.250,28	R\$32.502,80
No-break de 3,5 KVA	5	U	R\$5.500,00	R\$27.500,00
No-break senoidais inteligentes de 2 KVA entrada bivolt saída 110V	8	U	R\$2.384,95	R\$19.079,60
Abraçadeiras de Nylon	1800	U	R\$0,15	R\$270,00
Anilhas para identificação de cabo UTP, numéricas e alfabéticas	600	U	R\$0,06	R\$36,00
Mouse AT (Microsoft, Logitech ou Genius)	90	U	R\$12,65	R\$1.138,50
Mouse AT e PS/2 3 botões (Microsoft, Logitech ou Genius)	110	U	R\$13,30	R\$1.463,00
Mouse PS/2 (Microsoft, Logitech ou Genius)	110	U	R\$12,85	R\$1.413,50
Caixas de som amplificadas – genéricas	35	U	R\$31,68	R\$1.108,80

¹ A tabela apresentada mostra alguns dos produtos realmente postos em um contrato aberto, feito pela Brasil Telecom. Entretanto, nem todos os produtos do contrato estão presentes. Os preços unitários e as quantidades foram levemente alteradas, a pedido da Equipe de Obtenção da empresa.

Pilhas (baterias) de setup – genérica	40	U	R\$9,00	R\$360,00
Teclado AT – Padrão Brasileiro ABNT2	55	U	R\$22,58	R\$1.241,90
Teclado OS/2 – Padrão Brasileiro ABNT2	50	U	R\$24,36	R\$1.218,00
Cabos paralelos para impressoras	40	U	R\$10,97	R\$438,80
Caixa com tomada RJ45 CAT 5E com contato banhado em ouro com 50 microns/polegada	60	U	R\$26,64	R\$1.598,40
Filtro de linha com 4 tomadas	45	U	R\$35,65	R\$1.604,25
Conector RJ45 com contato banhado em ouro com película de 50 mocrons/polegada	40	U	R\$0,84	R\$33,60
Conectores ópticos SC/ST duplex AMP	35	U	R\$53,64	R\$1.877,40
Cabos para conexão de Hubs 3COM, part number 3C16420	1000	MT	R\$98,14	R\$98.140,00
Cabo de fibra óptica multimodo com 6 fibras	35	CX	R\$6,00	R\$210,00
Transceiver AUI – ST fêmea	1100	MT	R\$6,28	R\$6.908,00
Transceiver AUI – TP	30	CX	R\$105,24	R\$3.157,20
Punch Down para efetuar conexões em Patch Panel	25	U	R\$615,35	R\$15.383,75
Memória DIMM 128MB para Desktop PC-100 / PC-133	120	U	R\$159,35	R\$19.122,00
Memória DIMM 128MB para Notebooks Acer e Compaq	80	U	R\$320,48	R\$25.638,40
Memória DIMM 256MB IBM part Number 33L3075	100	U	R\$250,54	R\$25.054,00
Memória DIMM 256MB para Desktop PC-100 / PC-133	110	U	R\$274,15	R\$30.156,50
Memória DIMM 512MB para Desktop PC-100 / PC-133	100	U	R\$800,00	R\$80.000,00
Memória DIMM 512MB, 100Mhz, 3.3v, 4k Refresh ECC	55	U	R\$875,20	R\$48.136,00
Memória DIMM 64MB para Desktop PC-100 / PC-133	50	U	R\$85,34	R\$4.267,00
Memória DIMM ECC, kits de 256MB para RISC IBM 9076SP	5	U	R\$1.534,62	R\$7.673,10
Placa de Som – Genérica	20	U	R\$50,36	R\$1.007,20
Patch cords de 2 metros UTP, categoria 5E	30	U	R\$11,94	R\$358,20
Patch cords de 5 metros UTP, categoria 5E	10	U	R\$20,19	R\$201,90
Patch Panel 24 portas UTP Categoria 5	50	U	R\$415,00	R\$20.750,00
Patch Panel 48 portas UTP Categoria 5	60	U	R\$856,24	R\$51.374,40
Cordões ópticos SC/SC duplex multimodo, 2,5 metros	90	U	R\$105,83	R\$9.524,70
Cordões ópticos ST/ST duplex multimodo, 2,5 metros	110	U	R\$114,13	R\$12.554,30
TOTAL:				R\$786.219,71