

PAULO ANDRÉ LIMA DE CASTRO

**UMA INFRA-ESTRUTURA PARA AGENTES ARREMATANTES
EM MÚLTIPLOS LEILÕES SIMULTÂNEOS**

Dissertação apresentada à Escola Politécnica da
Universidade de São Paulo para obtenção do Título
de Mestre em Engenharia

São Paulo
2003

PAULO ANDRÉ LIMA DE CASTRO

**UMA INFRA-ESTRUTURA PARA AGENTES ARREMATANTES
EM MÚLTIPLOS LEILÕES SIMULTÂNEOS**

Dissertação apresentada à Escola Politécnica da
Universidade de São Paulo para obtenção do Título
de Mestre em Engenharia

Área de Concentração:
Sistemas Digitais

Orientador:
Prof. Dr.
Jaime Simão Sichman

São Paulo
2003

FICHA CATALOGRÁFICA

Castro, Paulo Andre Lima de

Uma infraestrutura para agentes inteligentes arrematantes em múltiplos leilões simultâneos / P.A.L. Castro. -- São Paulo, 2003.

p.104

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais.

1.Inteligência artificial 2.Agentes inteligentes 3.INTERNET
I.Universidade de São Paulo. Escola Politécnica.
Departamento de Engenharia de Computação e Sistemas

AGRADECIMENTOS

Ao meu orientador Prof. Dr. Jaime Simão Sichman pelas diretrizes claras e objetivas e por ter aceitado orientar este trabalho.

A Tuomas Sandholm e Kate Larson por facilitar o desenvolvimento deste trabalho, dirimindo dúvidas sobre o sistema eMediator.

À Isabel, minha esposa, e à Maria Eduarda, minha filha, pela paciência nos momentos de ausência e pelo incansável apoio.

Aos amigos Alexandre, Cômodo, Vani e Vinícius pela colaboração nas correções.

A todos que, direta ou indiretamente, colaboraram na execução deste trabalho.

RESUMO

Em meio a centenas de sítios de leilões na Internet, observa-se que alguns oferecem serviços de agentes para procurar, monitorar e/ou oferecer lances nos leilões. A maior parte dos sistemas baseados em agentes não oferece apoio ao acompanhamento e à submissão de lances em vários leilões simultâneos. Estas atividades são especialmente importantes, pois o fato de participar simultaneamente em vários leilões eleva a probabilidade de conseguir melhores resultados econômicos, devido à possibilidade de encontrar mercados mais vantajosos. Contudo, à medida que cresce o número de leilões simultâneos, torna-se inviável para um ser humano monitorá-los e oferecer lances. Neste contexto, o uso de agentes pode ser extremamente interessante. Neste trabalho, propõe-se uma infra-estrutura para facilitar a construção de agentes arrematantes para operarem no eMediator (Sandholm, 2000) em múltiplos leilões. Com tal infra-estrutura computacional batizada de sistema AAS (Auction Agent System), a implementação de novas estratégias em agentes arrematantes será facilitada. Desta forma, este trabalho contribui para o avanço na pesquisa de novas estratégias de atuação de agentes arrematantes para o caso de múltiplos leilões simultâneos.

ABSTRACT

Among hundred of auction sites in the internet, some of them offer services powered by software agents that search auctions, monitor them and/or submit bids in these auctions. However, most of these systems are concerned about just one auction, instead of acting in multiple simultaneous auctions. The possibility of acting in multiple auctions is very important, because it makes possible to get better economical results. But if the number of auctions grows very much, it can become very hard (or even impossible) to a human being to monitor and bid in these auctions. That can make the use of software agents very useful. We propose a infrastructure that makes easier the construction of bidder agents that works with eMediator (Sandholm, 2000) system and new bidding strategies. We call this infrastructure AAS (Auction Agent System). This system adds value to researchers of multiple simultaneous auctions bidding strategies.

LISTA DE FIGURAS

Figura 3.1. Escopo de Agentes Inteligentes.....	24
Figura 3.2. Taxonomia de Agentes por Franklin e Graesser.....	28
Figura 3.3. Sistemas cooperativos com agentes distribuídos.....	31
Figura 4.1. Arquitetura do AuctionBot.....	37
Figura 4.2. Arquitetura do BiddingBot.....	40
Figura 4.3. Arquitetura do AgILE.....	41
Figura 4.4. Arquitetura da eAuctionHouse.....	44
Figura 5.1. Exemplo de atuação de agentes AAS.....	48
Figura 5.2. Arquitetura de Software do AAS.....	51
Figura 5.3. Herança de algumas classes no AAS e de alguns agentes arrematantes.....	53
Figura 5.4. Diagrama de Atividades do Agente arrematante na AAS.....	58
Figura 5.5. Apresentação do desempenho de um agente arrematante para o usuário.....	59
Figura 5.5. Funcionamento do AAS.....	60
Figura 5.7. Tela de configuração do agente Greedy.....	64
Figura 5.8. Tela de configuração do agente Remaining Time (RT)	64
Figura 6.1. Atuação do Agente Greedy	67
Figura 6.2. Atuação do Agente RT.....	69
Figura 6.3. . Comparação de Atuação dos Agentes Greedy e RT.....	71

LISTA DE TABELAS

Tabela 3.1. Características comuns em Agentes Autônomos.....	23
Tabela 4.1. Comparação entre Sistemas.....	45
Tabela 5.1. Agentes AAS e suas características essenciais.....	49
Tabela A.1. Sintaxe dos Comandos AAS (BNF)	80
Tabela A.2. Caracteres Especiais e seqüências de Escape nos comandos AAS.....	80
Tabela A.3. Lista de Comandos AAS.....	81
Tabela A.4. Parâmetros dos Comandos AAS.....	82
Tabela A.5. Parâmetros de uma Lista de Leilões e de uma lista de lances.....	83
Tabela A.6. Parâmetros dos Lances no AAS.....	84
Tabela A.7. Parâmetros de um Leilão no AAS.....	85
Tabela B.1. Log do Agente “GreedyAgent” – Experimento 1.....	91
Tabela B.2. Log do Agente “GreedyAgent1” – Experimento 1.....	92
Tabela B.3. Log do Agente “GreedyAgent2” – Experimento 1.....	94
Tabela B.4. Log do Agente “GreedyAgent3” – Experimento 1.....	95
Tabela B.5. Log do Agente “RTAgent” – Experimento 2.....	97
Tabela B.6. Log do Agente “RTAgent1” – Experimento 2.....	98
Tabela B.7. Log do Agente “GreedyAgent” – Experimento 3.....	99
Tabela B.8. Log do Agente “RTAgent” – Experimento 3.....	100

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

1. INTRODUÇÃO	1
1.1. Objetivos	2
1.2. Escopo	4
1.3. Resultados Esperados	5
1.4. Avaliação dos Resultados	6
1.5. Organização do Documento	6
2. INTRODUÇÃO À TEORIA DE LEILÕES	7
2.1. Introdução	7
2.2. Conceitos Básicos	8
2.3. Mecanismo de Determinação de Preço e Fechamento	9
2.4. Iniciativa do Lance	10
2.5. Os Tipos Básicos de Leilões	12
2.6. Estratégias de Atuação	14
2.7. Leiloando Múltiplos Itens	15
2.8. Limitações no Uso de Leilões como Mecanismo de Negociação	16
2.9. Leilões Eletrônicos	17
2.10. Atuação em Múltiplos Leilões Simultâneos	17
2.11. Conclusões	18
3. AGENTES AUTÔNOMOS : UMA VISÃO DA ÁREA	19
3.1. Introdução	19
3.2. Algumas definições de Agentes Autônomos	20
3.2.1. Abordagem Atributiva	20
3.2.2. Abordagem Descritiva	22
3.3. Taxonomia de Agentes Autônomos	27
3.4. Motivações para a construção de Agentes	29
3.4.1. Simplificando a Computação Distribuída	29
3.4.2. Superando Limitações em Interfaces com o Usuário	31
3.5. Sociedades de Agentes e Sistemas Multiagentes	34
3.6. Conclusões	35
4. SISTEMAS DE LEILÕES BASEADOS EM AGENTES	36
4.1. AuctionBot	37
4.2. BiddingBot	39
4.3. AgILE	41
4.4. eMediator	43
4.5. Análise Comparativa	45

4.6. Conclusões	46
5. PROJETO DO SISTEMA AAS	47
5.1. Agentes no AAS	47
5.2. Arquitetura do AAS	50
5.3. O agente de busca de leilões: AuctionSearcher	54
5.4. O agente de monitoramento de leilões: AuctionMonitor	55
5.5. O agente arrematante: Bidder	56
5.6. Funcionamento do AAS	60
5.7. Estratégias de Atuação dos Agentes no AAS	62
5.8. Conclusões	65
6. EXPERIMENTOS E AVALIAÇÃO DE RESULTADOS	66
6.1. Atuação do Agente Greedy	67
6.2. Atuação do Agente RT	68
6.3. Comparação de Atuação dos Agentes Greedy e RT	71
6.4. Avaliação dos Resultados	73
6.5. Conclusões	74
7. CONCLUSÕES	75
7.1. Resultados Obtidos	75
7.2. Trabalhos Futuros	77
7.3. Considerações Finais	79
ANEXO A - Comandos AAS	80
ANEXO B - Experimentos Realizados	90
LISTA DE REFERÊNCIAS	101

1. INTRODUÇÃO

Existem hoje muitos sistemas comerciais de leilão na Internet que implementam os mais variados tipos de leilões, cada um com suas próprias particularidades. Embora haja uma grande quantidade de pesquisas sobre este tema, o uso de agentes de software para automatizar as tarefas inerentes à participação em um leilão é relativamente recente e conta com um número reduzido de sistemas (Sandholm, 2000).

Entre os vários sistemas eletrônicos de leilões, observa-se a existência de algumas casas de leilão acadêmicas inovadoras, por serem baseadas em agentes. Dentre os sistemas possíveis foram selecionados três sistemas: AuctionBot (Wurman; Wellman; Walsh, 1998), o eMediator (Sandholm, 2000) e BiddingBot (Ito; Fukuta; Sycara, 2000). Embora a maior parte das pesquisas concentre-se no projeto de agentes que atuam em um único leilão por vez, apesar destas pesquisas sem dúvida serem importante, acredita-se que o uso de agentes capazes de atuar simultaneamente em múltiplos leilões crescerá fortemente, pois pode criar mercados mais eficientes (Preist; Bye; Bartolini, 2001).

A maior parte dos sistemas baseados em agentes não oferece apoio ao acompanhamento e à submissão de lances em vários leilões simultâneos, sejam do mesmo tipo ou de tipos diferentes. Estas atividades são especialmente importantes, pois o fato de participar (monitorar e eventualmente dar lances) simultaneamente de vários leilões eleva a probabilidade de conseguir melhores resultados econômicos, devido à possibilidade de encontrar mercados mais vantajosos. Contudo, à medida que cresce o número de leilões simultâneos, torna-se difícil para um ser humano monitorá-los e oferecer lances.

Neste contexto, o uso de agentes pode ser extremamente interessante, já que com o aumento do número de leilões monitorados cresce a probabilidade de se obter maior economia para o proprietário do agente. Na realidade, existem algumas pesquisas para obtenção de algoritmos ótimos para atuação em múltiplos leilões, o que envolve escolher em que leilão oferecer lance e qual o valor do lance (Anthony et al., 2001) (Byde; Preist; Jennings, 2002) (Preist; Bartolini; Philips, 2001). Todavia, tais estratégias ainda não foram implementadas em servidores de leilões eletrônicos. Em Anthony et al. (2001), há a simulação da atuação do agente projetado e a obtenção de alguns resultados empíricos. Entretanto, trata-se apenas de experimentos em um ambiente simplificado, que não abordam os vários aspectos necessários para implementar o agente em um servidor de leilões real. O objetivo deste trabalho é efetuar esta implementação usando como base o eMediator, em especial seu módulo de leilões denominado eAuctionHouse (Huai; Sandholm, 2000). Desta forma, o presente trabalho é um dos pioneiros, no que tange a implementação de alguns destes algoritmos num ambiente real de leilões eletrônicos.

1.1. Objetivos

O presente trabalho tem como objetivo desenvolver uma infra-estrutura para facilitar a construção de agentes arrematantes para operarem no eMediator em múltiplos leilões. Com tal infra-estrutura computacional, a implementação de novas estratégias e a comparação dos resultados obtidos pelos agentes são facilitadas. Adicionalmente, pretende-se desenvolver dois agentes que atuem em múltiplos leilões, cada um com uma estratégia de operação diferente para validar e comprovar tais funcionalidades. Além disso, os resultados obtidos por estes agentes são comparados através de gráficos de atuação. Os produtos decorrentes deste trabalho são os seguintes:

- **AAS (Auction Agent System):** Trata-se de um sistema que engloba uma biblioteca de classes escrita em Java, que encapsula a API (*Application Programm Interface*) do eMediator e agentes que trabalham em conjunto para buscar e monitorar leilões, além de oferecer lances nestes leilões. A API do eMediator é baseada em cadeias de caracteres formatadas e transferidas através de conexões TCP/IP com o servidor, o que exige um razoável conhecimento da implementação do servidor e dificulta o desenvolvimento de agentes. A biblioteca do AAS irá oferecer uma interface de programação orientada a objetos e com nível de abstração mais alto, evitando assim que o projetista de agentes tenha que se preocupar com detalhes de implementação do eMediator. O AAS engloba ainda dois agentes, que realizam para o agente arrematante as atividades de busca e monitoramento dos leilões de interesse, facilitando assim ainda mais o desenvolvimento de agentes arrematantes. Estes agentes e a biblioteca citada constituem uma infra-estrutura para a construção de novos agentes arrematantes.
- **Agentes Arrematantes:** Utilizando o AAS, serão desenvolvidos dois agentes que atuem em múltiplos leilões simultâneos com tempos de fechamento distintos e não sincronizados. As estratégias a serem implementadas por estes agentes são as seguintes: **Greedy** (Byde, 2001) e **RT** (Anthony et al., 2001), que serão detalhadas na seção 5.3.

Não se pretende neste trabalho definir estratégias ótimas para atuar em leilões múltiplos, uma vez que a complexidade de tal tarefa fugiria ao escopo de um trabalho de mestrado. O considerável número de artigos relacionados ao tema comprova as enormes dificuldades de determinar uma estratégia ótima para leilões múltiplos (Anthony et al., 2001) (Byde; Preist; Jennings, 2002) (Preist; Bartolini; Philips, 2001). Entretanto, com o desenvolvimento de um

sistema que facilite a implementação de agentes que operem em um servidor de leilões real (eMediator), simplifica-se a determinação de algoritmos mais eficientes.

1.2. Escopo

Para visualizar com maior clareza algumas limitações deste trabalho, é necessário entender melhor as fases envolvidas em um processo de compra. Existem vários modelos que descrevem tal processo. Entretanto, todos eles contemplam seis fases comuns, previstas no modelo CBB (*Consumer Buying Behaviour*) como citado por Maes, Guttman e Moukas (1999). Cada fase descreve um conjunto de atividades realizadas pelo potencial comprador. Estas fases são as seguintes:

1. **Identificação de Necessidade:** Nesta fase, o consumidor percebe uma necessidade não satisfeita, independentemente de informações a respeito de produtos específicos, e define tal necessidade.
2. **Procura por Produto:** Aqui o consumidor compara informações dos vários produtos disponíveis e seleciona um grupo de produtos, chamado de conjunto de consideração (*consideration set*), que atenderiam suas necessidades.
3. **Procura por Fornecedor:** Neste estágio, combina-se o conjunto de consideração (produtos definidos na fase anterior) com informações específicas a respeito dos possíveis fornecedores, como por exemplo: garantia, disponibilidade de entrega, reputação, preço de referência, etc. Este conjunto de informações irá subsidiar a fase seguinte.

4. **Negociação:** Nesta fase, determinam-se os termos da transação. O processo de negociação pode variar bastante em termos de tempo de duração e complexidade, de acordo com a forma de negociação adotada.
5. **Compra e Entrega:** Trata-se da execução do que foi acordado na fase de negociação.
6. **Serviços ao Consumidor e Avaliação:** Nesta fase, incluem-se serviços ao consumidor após a venda, assistência técnica e a avaliação da satisfação do processo de compra como um todo. A natureza desta fase pode variar bastante de acordo com o tipo de produto comercializado.

O sistema proposto destina-se a abordar as fases 3 e 4, procura por fornecedor e negociação, respectivamente. A fase de procura por fornecedor é feita através da busca de vários leilões, cada leilão representa na realidade um fornecedor em potencial, e o usuário não precisa se preocupar em buscar informações a respeito dos fornecedores. A negociação se dá através de leilão, o qual pode ser inglês, holandês, leilão de Vickrey ou de lances fechados e primeiro preço, ou seja os tipos contemplados pelo sistema. As definições dos vários tipos de leilões são apresentadas no capítulo 2.

1.3.Resultados Esperados

Como resultado principal do trabalho tem-se um sistema (Auction Agent System, AAS) para facilitar a construção de agentes para operarem no eMediator (Sandholm, 2000) (Huai; Sandholm, 2000). Além disso, serão implementados dois agentes, que utilizam diferentes estratégias de operação, para demonstrar a utilidade do próprio AAS. Tais estratégias são

encontradas na literatura especializada (Anthony et al., 2001) (Byde, 2001). Tanto o AAS como os agentes desenvolvidos podem ser posteriormente utilizados para implementar agentes de leilões com estratégias distintas e avaliar seus desempenhos, facilitando assim o trabalho de criar e otimizar estratégias para atuação de agentes em leilões múltiplos e até mesmo em leilões únicos, uma vez que o sistema pode também ser utilizado para construir agentes para este último caso.

1.4. Avaliação dos Resultados

A avaliação do sistema AAS levará em consideração a diminuição de esforço proporcionada ao projetista de agentes arrematantes. Demonstrar que um modo de executar uma determinada atividade exige menos esforço que outro, nem sempre é uma tarefa simples. Entretanto, pode-se identificar uma diminuição do esforço à medida que tarefas antes sob a responsabilidade do projetista de agentes, passem a ser realizadas pelo próprio sistema.

1.5. Organização do Documento

O capítulo 2 apresenta uma introdução à teoria de leilões, a qual constitui, juntamente com a área de agentes autônomos, a base fundamental para este trabalho. Uma introdução a agentes autônomos é exibida no capítulo 3. O capítulo 4 descreve alguns sistemas de leilões eletrônicos existentes baseados em agentes autônomos, fazendo uma comparação entre tais sistemas. Em seguida, no capítulo 5 descreve-se a arquitetura do sistema, seus agentes e suas características de funcionamento. Finalmente, no capítulo 6 são demonstrados os resultados obtidos em um conjunto de experimentos e uma avaliação destes resultados. As conclusões do trabalho, bem como algumas perspectivas futuras, são apresentadas no capítulo 7.

2. INTRODUÇÃO À TEORIA DE LEILÕES

Neste capítulo, será introduzida a Teoria de Leilões como um ramo da teoria econômica. Os conceitos básicos da área são descritos na seção 2.2. Os vários mecanismos para determinar o fechamento do leilão, o vencedor e o preço a ser pago por este são exibidos na seção 2.3. Em seguida, explica-se as variações possíveis quanto à iniciativa do lance na seção 2.4 e, na seção 2.5, descreve-se os tipos de leilões mais comuns. Nas seções 2.6, aborda-se alguns conceitos envolvidos na formulação de estratégias de atuação, enquanto na seção 2.7, trata-se da questão de leiloar múltiplos itens. Ao final do capítulo, são apresentadas limitações no uso de leilões (seção 2.8), o advento de leilões em sítios na Internet (seção 2.9), algumas considerações sobre a atuação simultânea em múltiplos leilões (seção 2.10) e conclui-se o capítulo na seção 2.11.

2.1. Introdução

Dentro da Teoria Econômica, um ramo relativamente novo trata do estudo de leilões e suas aplicações. Tal ramo, denominado Teoria de Leilões (Klemperer, 1999), tem ganhado uma importância crescente devido a razões práticas, empíricas e teóricas. Para uma introdução à Teoria de Leilões, uma excelente referência é Klemperer (1999), onde o assunto é apresentado em linguagem simples e acessível para iniciantes e onde é indicado também um conjunto dos principais trabalhos da área econômica relacionados ao tema. A Teoria de Leilões se impõe como um assunto de grande importância, em primeiro lugar, porque um enorme volume de recursos é transacionado atualmente através de leilões. Em vários países, nos últimos anos, utilizou-se de leilões para privatizar estatais, vender concessões de

telefonia fixa e móvel, direitos de exploração de petróleo, etc. No Brasil, várias empresas estatais e também concessões de exploração de serviços de telefonia celular e fixa foram vendidos através de leilões, envolvendo valores na ordem de alguns bilhões de reais. Contratos governamentais ou de empresas de grande porte são tipicamente negociados através de leilões reversos (*procurement auctions*). Neste caso, o leiloeiro está buscando o lance de menor preço, ao contrário do caso comum (leilão direto), onde o objetivo do leiloeiro é obter o maior preço possível pelo item que está sendo leiloado. Além disso, imóveis, gado, produtos agrícolas, peças de arte e antiguidades são muitas vezes vendidos através de leilões. Um exemplo é o caso das bolsas de mercadorias, onde commodities são negociadas através de leilões duplos, que se caracterizam pelo fato de comprador e vendedor oferecerem lances que são associados (um lance comprador associa-se a um lance vendedor) de modo conveniente, definindo uma transação de negócio.

Além disso, como leilões são simples e bem definidos, estes podem ser utilizados para testar hipóteses, especialmente na área de Teoria dos Jogos. A teoria de leilões tem sido utilizada para uma série de estudos em assuntos como formação de preços, teoria de preço monopolístico e outros (Klemperer, 1999).

2.2. Conceitos Básicos

Para um melhor entendimento da Teoria de Leilões, um pequeno grupo de conceitos básicos é importante. Nesta seção, são apresentados tais conceitos e alguns termos comuns que são utilizados na área.

- **Leiloeiro:** Pessoa ou entidade responsável pelo andamento do leilão. Geralmente, um terceiro que não está comprando ou vendendo produtos nos leilões, mas se

coloca a serviço do proprietário do bem a ser leilado e dos potenciais compradores para facilitar a negociação através do leilão.

- **Arrematante:** Pessoa ou entidade que oferece lances em um leilão. Em geral, trata-se do potencial comprador, porém em leilões reversos (descritos na seção 2.8) o arrematante submete lances com o intuito de vender seus produtos. Nos textos em inglês, utiliza-se o termo “*Bidder*”.
- **Preço Reservado:** Maior valor que um determinado arrematante de um leilão está disposto a pagar para obter o bem em questão. Tal informação, via de regra, não é tornada pública pelo arrematante, pois comprometeria seu objetivo de obter o bem pagando um preço menor.

2.3. Mecanismo de Determinação de Preço e Fechamento

As estratégias de atuação em um leilão são fortemente dependentes das regras do leilão, especialmente das regras que definem o fechamento e a determinação do vencedor e do preço que será pago por este. Os leilões podem ser agrupados quanto aos mecanismos de fechamento do seguinte modo:

- **Fechamento Temporal:** O leilão é fechado em um momento pré-determinado pelo leiloeiro. A data e hora do fechamento, em geral, são conhecidas por todos os arrematantes. Pode ser utilizado em praticamente qualquer tipo básico de leilão: inglês, americano (lances fechados e primeiro preço) e de Vickrey.
- **Fechamento por Evento:** O leilão é fechado por ocorrência de um evento definido pelo leiloeiro. Os eventos mais comuns utilizados como condição para o

fechamento são os seguintes: (i) apresentação de um lance (caso do leilão holandês), (ii) ordem direta do leiloeiro para fechar o leilão e (iii) fim de um determinado período sem apresentação de novos lances. Os dois últimos casos podem ser em leilões dos tipos inglês, americano (lances fechados e primeiro preço) ou de Vickrey, inclusive em conjunto com fechamento temporal, como regra adicional para fechar o leilão antes do prazo previsto.

Por outro lado, quanto à determinação de preço, tem-se basicamente dois tipos de leilões:

- **Primeiro Preço:** O arrematante paga o preço do seu próprio lance. É o caso mais comum e é utilizado em leilões do tipo inglês, holandês e americano.
- **Segundo Preço:** O arrematante paga o preço do lance do primeiro perdedor, isto é, do arrematante que deu o segundo maior lance. É utilizado principalmente para estudos devido a propriedades teóricas interessantes (Klemperer, 1999, p. 5).

2.4. Iniciativa do Lance

Uma das possíveis variações em todos os leilões é observada quando, ao invés de existir um vendedor e vários possíveis compradores que submetem os lances, tem-se um comprador e vários possíveis vendedores e estes ficam com o trabalho de submeter lances. Geralmente, nomeia-se tal caso de leilão reverso (*procurement auctions* ou *reverse auctions*).

Leilões reversos englobam também a situação geralmente tratada como compra por pedido de cotação (*Request For Quote*, RFQ), que pode ser vista como um leilão reverso de lances fechados e primeiro preço. Vários sítios de leilão atuam com leilões reversos para comercialização dos mais variados itens, de passagens aéreas à compra de insumos

industriais (como exemplos têm-se os sítios Priceline.com, Netchemistry.com, entre outros). Entretanto, o leilão reverso não é formalmente diferente do leilão normal, além da óbvia diferença que, neste caso, vence o menor preço e não o maior. Por isso, qualquer estratégia de atuação (aprofunda-se o assunto de estratégias de atuação na seção 2.6) definida para o caso normal vale também para o leilão reverso (Klemperer, 1999).

Existe ainda uma outra classe de leilão, quando a iniciativa do lance é de ambos - comprador e vendedor. Geralmente, esta classe de leilão é denominada de leilões duplos (*double auctions*) e é utilizado em bolsas de valores. Naturalmente, exige um mecanismo de determinação de preço diferente, pois há lances de compra e de venda. Via de regra, o preço pode ser definido de três formas, como:

- **Valor Médio:** Defini-se como a média entre um lance de compra e um lance de venda compatíveis entre si (quanto ao item e à quantidade de itens).
- **Valor do Lance de Compra:** O preço é definido como o valor do lance de compra. Neste caso, o vendedor pode receber um valor maior que seu próprio lance.
- **Valor do Lance de Venda:** O preço é definido como o valor do lance de venda. Neste caso, o comprador pode pagar um valor menor que seu próprio lance.

Caso o preço se estabeleça pelo lance de compra, só é realizado negócio quando existe um lance de venda com valor igual ou menor ao de compra, no caso de ser menor o vendedor não deixa de realizar o negócio, pois recebe um valor maior que o esperado por seu produto. Desta forma, o vendedor tenderia a ser beneficiado. Entretanto, a inversão de papéis entre

comprador e vendedor é característica de leilões duplos. Ao comprar um item, o comprador coloca-o novamente à venda passando a atuar no papel inverso. Raciocínio análogo vale quando se segue a regra de pagar o valor definido pelo lance de venda.

2.5. Os Tipos Básicos de Leilões

Os tipos de leilões mais utilizados são os seguintes (Anthony et al., 2001) (Klemperer, 1999):

- **Leilão Inglês:** Também chamado de aberto, oral ou leilão de lances ascendentes. Neste caso, os arrematantes dão lances crescentes para um determinado item até que nenhum arrematante esteja disposto a dar um lance maior que o atual. É provavelmente o tipo mais comum de leilão. Este leilão pode ser feito com o vendedor anunciando, opcionalmente, um preço mínimo. Os arrematantes oferecem seus lances, cujos valores podem ser públicos, ou apenas o melhor lance terá seu valor publicado. Há ainda outra variação possível, onde se apresentam todos os lances, exceto o lance vencedor. Este tipo de leilão é muito utilizado para vender obras de arte ou antiguidades, por exemplo. Na Internet, este o tipo de leilão é o mais utilizado atualmente por meio do qual são vendidos os mais variados tipos de produtos.
- **Leilão Holandês:** Funciona de modo inverso ao leilão inglês. O leiloeiro fixa um preço inicial alto e vai progressivamente diminuindo tal preço, até que algum dos arrematantes esteja disposto a comprar o item pelo preço corrente. É assim chamado devido ao fato de ser utilizado por vendedores de flores na Holanda. Neste caso, os potenciais compradores ficam em uma sala onde sentam em mesas

com painéis ligados a um relógio eletrônico na frente da sala. O relógio apresenta em seu interior o que está sendo leiloado (uma certa quantidade de flores) e o preço inicial do leilão. Quando se inicia o leilão uma série de luzes ao redor do relógio indica o preço corrente através da redução do percentual do preço original do bem. Quando um comprador aciona seu painel, ele compra as flores pelo preço indicado pelo relógio. Caso haja vários lotes de flores em leilão, o comprador pode escolher apenas alguns dos lotes e os restantes são leiloados novamente. No Canadá, se leiloa tabaco, e em Israel se leiloa peixe de modo semelhante.

- **Leilão de Lances Fechados e Primeiro Preço:** Os arrematantes entregam envelopes lacrados com seu respectivo lance independentemente, e cada arrematante tem direito a apresentar apenas um lance. Os envelopes são abertos juntos. O bem é vendido para o maior lance e o vencedor paga o valor estipulado pelo próprio lance. O preço é então definido pelo maior ou “primeiro” lance. Leilões deste tipo foram usados em algumas privatizações brasileiras (por exemplo, a privatização do Banco do Estado de São Paulo, BANESPA). O leilão de lances fechados e primeiro preço é também chamado por alguns de **Leilão Americano** (Garcia; Lopes; Bentes, 2001).
- **Leilão de Vickrey:** Também é chamado de Leilão de Lances Fechados e Segundo Preço. É similar ao leilão de lances fechados e primeiro preço, com a diferença de que o valor pago pelo vencedor é o segundo maior lance dado no leilão, ou seja, o vencedor paga o valor definido pelo primeiro perdedor. Foi criado por William Vickrey em 1961 (Klemperer, 1999). É menos comumente

utilizado que os outros três tipos, porém é muito estudado por apresentar propriedades teóricas interessantes, como será detalhado na seção 2.6.

2.6. Estratégias de Atuação

A Teoria dos Jogos fornece uma fundamentação teórica importante para as estratégias de atuação em leilões e é utilizada em alguns sistemas eletrônicos de leilão, como por exemplo, o AgILE (Garcia; Lopes; Bentes, 2001). Dois conceitos fundamentais na Teoria de Jogos são os seguintes:

- **Estratégia Dominante:** Uma estratégia é dominante (ou completamente dominante) quando independentemente dos lances dos outros jogadores (ou arrematantes do leilão) é a melhor estratégia a ser seguida para maximizar a utilidade para o jogador. Em leilões, a utilidade para o arrematante será função da diferença entre seu preço reservado e o valor efetivamente pago pelo bem leilado e da probabilidade de obter o produto desejado. Entretanto, arrematantes do leilão podem levar em consideração outros fatores, como, por exemplo, o tempo despendido para a compra do item. Quando da formulação da estratégia, deve-se levar em consideração a função de utilidade que se deseja maximizar.
- **Estratégia Dominada:** Uma estratégia dominada (ou completamente dominada) é aquela onde qualquer que seja o conjunto de lances dos demais jogadores, sempre há uma estratégia melhor a ser seguida (Green; Whinston; Mas-Colell, 1995).

No leilão inglês, a estratégia dominante é dar como lance um valor mínimo acima do lance corrente até vencer o leilão ou ser atingido o valor de reserva, o que é facilmente demonstrável. Já no caso do leilão de Vickrey a estratégia dominante é pura e simplesmente dar um lance com valor igual ao preço reservado (Klemperer, 1999). Entretanto, em grande parte dos casos não é possível determinar uma estratégia dominante, o que leva ao uso de outras técnicas e abordagens fora da Teoria dos Jogos.

2.7. Leiloando Múltiplos Itens

Ao se leiloar vários itens, podem-se utilizar variações dos leilões dos tipos básicos (descritos na seção 2.5) ou então usar um outro tipo de leilão denominado **Leilão Combinatório**. Para utilizar os tipos básicos, basta colocar o lote de produtos em leilão e aceitar lances parciais (isto é, para um subconjunto do lote). Ao ser arrematado um subconjunto do lote, reinicia-se o leilão com os produtos remanescentes. O leilão combinatório permite que o arrematante defina subconjuntos dos itens que estão em leilão e submeta lances para um ou mais destes subconjuntos. Porém, os lances vencedores e os sublotes só são definidos ao final do leilão, por necessidade de considerar todos os lances ofertados e a possibilidade de atender a todos ou à parte deles, buscando obter o maior retorno possível para o leiloeiro. Pode ser utilizado quando há preferências do comprador por um subconjunto dos itens em detrimento do total de itens. Em leilões combinatórios, tem-se um modo economicamente eficiente de comprar ou vender combinações de vários itens. Entretanto, por exigir uma estrutura centralizada para coordenar este tipo de leilão, é dificilmente criado na prática, pois os bens a serem leiloados, geralmente pertencem a diferentes proprietários (Preist; Byde; Bartolini, 2001). Para maiores referências, veja (Sandholm, 2000).

2.8. Limitações no Uso de Leilões como Mecanismo de Negociação

O principal objetivo de uma negociação, independentemente da forma que se venha a adotar, é chegar a um acordo satisfatório para todos os participantes. Tendo em vista tal objetivo, um mecanismo de negociação pode ser classificado como colaborativo (*Integrative*) ou distributivo (*Distributive*)

- **Negociações Colaborativas:** Ocorrem quando uma alteração nos termos da negociação pode ser benéfica a todas as partes. Um exemplo é a situação onde um cliente aceita um prazo maior de entrega em troca de um preço menor e isto também é aceito pelo fornecedor, que está com sua capacidade de entrega saturada no curto prazo mas não deseja perder o cliente. Para uma negociação colaborativa é fundamental a existência de pelo menos duas características em discussão (no exemplo, prazo de entrega e preço). São também conhecidas como negociações “ganha-ganha”.
- **Negociações Distributivas:** Neste caso, qualquer alteração nos termos da negociação é prejudicial a uma das partes. Ocorre, por exemplo, em leilões onde se negocia apenas sobre o preço. Um aumento no preço é benéfico para o vendedor e prejudicial para o comprador e vice-versa. Uma negociação colaborativa pode se reduzir a uma distributiva pela fixação de todos os termos do negócio exceto um. Da mesma forma, uma negociação distributiva pode tornar-se colaborativa pela consideração de mais características. Às vezes, são chamadas de negociações “perde-ganha”.

É fácil perceber que um leilão é sempre uma negociação distributiva por tratar exclusivamente de uma variável, o preço. Entretanto, tal restrição pode ser superada através do uso de leilões multiatributos (Guttman; Maes; Moukas, 1998). Estes se baseiam em funções de utilidade definidas a partir das preferências do consumidor e das características relevantes do(s) produto(s) avaliado(s). Além de garantir uma outra categoria de negociação (a negociação colaborativa) é possível abordar o problema de negociações nas quais os produtos não são escolhidos apenas pelo preço.

2.9. Leilões Eletrônicos

A Internet popularizou o uso de leilões como meio de negociação e venda dos mais variados produtos. São os chamados leilões “*on-line*” ou leilões eletrônicos. Segundo Anthony et al. (2001), em 1998 já havia mais de 500 sítios de leilões na Internet e, provavelmente, este número cresceu de forma significativa até hoje. Segundo dados da Forrester Research 1999, foram transacionados US\$ 1,4 bilhões em 1998 através de leilões. Estima-se ainda que se alcançará o valor de US\$ 19 bilhões a serem transacionados em 2003. Este aumento no número de leilões e de produtos leiloados fez crescer também o problema de encontrar e realizar o melhor negócio para satisfazer uma necessidade do cliente. Entretanto, sob o ponto de vista do comprador, aumenta também a possibilidade de obter melhores negócios ao se participar de vários leilões simultaneamente.

2.10. Atuação em Múltiplos Leilões Simultâneos

Apesar de haver uma considerável quantidade de pesquisas em projeto de agentes e estratégias para atuação em leilões eletrônicos, a maior parte destes trabalhos concentra-se

no caso de um único leilão. Não obstante estes trabalhos serem sem dúvida muito importantes, acredita-se que, em longo prazo, o caso de atuação em leilões múltiplos simultâneos deverá sobressair-se, pois este pode criar mercados mais eficientes e estáveis (Preist; Byde; Bartolini, 2001).

2.11. Conclusões

Este trabalho se propõe a criar uma infra-estrutura computacional baseada em agentes para atuar em múltiplos leilões simultâneos, contemplando os quatro tipos básicos de leilões (inglês, holandês, americano e de Vickrey) e diferentes mecanismos de fechamento. Serão implementados ainda agentes arrematantes com estratégias de atuação distintas. Os principais conceitos referentes são descritos no capítulo 3.

3. AGENTES AUTÔNOMOS : UMA VISÃO DA ÁREA

Neste capítulo, apresenta-se uma visão geral da área de agentes autônomos. Na seção 3.1, introduz-se o assunto e destaca-se a diversidade de linhas de pesquisas abrangidas e a variedade de definições do termo agente. Na seção 3.2, são descritas algumas definições de agentes, salientando-se a definição que será utilizada neste trabalho. Em seguida, apresenta-se a taxonomia de agentes que será adotada, na seção 3.3. Finalmente, as motivações para a construção de agentes são discutidas (seção 3.4), são exibidos alguns conceitos de Inteligência Artificial Distribuída (IAD) que serão utilizados ao longo deste trabalho (seção 3.5) e destacados os principais assuntos do capítulo (seção 3.6).

3.1. Introdução

A idéia de construir agentes artificiais fascina a humanidade há muito tempo. Robôs, ciborgues, andróides são freqüentemente usados em obras de ficção científica e encantam multidões. Mais recentemente, surgiu a idéia de robôs (ou agentes) que existem apenas dentro de um computador, os agentes de software, também chamados de “*SoftBots*”. Estes agentes não teriam estruturas mecânicas próprias para realizarem ações como os robôs, entretanto as tarefas que podem desempenhar são igualmente diversas. Na realidade, o assunto de agentes autônomos que engloba o conceito de agente de software é estudado por grande número de pesquisadores com as mais diferentes origens e das mais diferentes áreas, como, por exemplo: inteligência artificial distribuída, robótica, vida artificial, objetos distribuídos, interação homem-máquina, interfaces inteligentes e adaptativas entre outras (Bradshaw, 1997). Esta diversidade tem trazido novas aplicações e novas abordagens a cada dia, porém trouxe também uma enorme quantidade de interpretações diferentes para alguns

termos utilizados. Por exemplo, alguns programas são chamados de agentes simplesmente por realizarem atividades agendadas em uma máquina remota; outros, por implementarem uma primitiva ou agregado de uma função cognitiva, outros por desempenharem o papel de um assistente pessoal (Bradshaw, 1997). Esta variedade de interpretações gera um significativo grau de confusão. Na seção 3.1, são apresentadas algumas propostas de definições para o conceito de agente, incluindo a definição que será seguida no decorrer deste trabalho. Na seção 3.2, é apresentada uma taxonomia de agentes autônomos. Em seguida, apontam-se algumas aplicações da tecnologia de agentes (seção 3.3) e, em especial, são destacadas algumas aplicações na área de comércio eletrônico (seção 3.4).

3.2. Algumas definições de Agentes Autônomos

Segundo Bradshaw (1997), duas abordagens complementares podem ser utilizadas para tentar encontrar uma definição de agente: atributiva e descritiva. O primeiro caso tem como base a atribuição feita por alguém para o agente. Já o outro caso baseia-se em uma descrição dos atributos que uma entidade deve apresentar de modo a ser classificada como um agente. Estas abordagens serão apresentadas nas seções 3.2.1. e 3.2.2.

3.2.1. Abordagem Atributiva

Na abordagem atributiva, o aspecto principal do agente é o que ele faz sob o ponto de vista do seu proprietário. Ao invés de listar uma série de atributos, o agente é definido a partir das atividades que pode desempenhar em nome de seu proprietário e em que grau de autonomia as realiza. Como afirma Bradshaw, “o agente é o que o agente faz” e complementa: “a essência da visão de agência não pode ser caracterizada em última instância, listando uma

coleção de atributos; ao invés disso, consiste fundamentalmente de uma atribuição feita por parte de uma pessoa”.

A afirmação de Bradshaw ajuda a compreender a variedade de interpretações do que poderia ser definido como um agente. Para alguns, aquilo que seria um agente inteligente para outros é apenas um programa tradicional. Da mesma forma, o agente inteligente de hoje pode se tornar o programa tradicional de amanhã. Bradshaw observa que a característica chave para fazer esta distinção são as expectativas e experiências sob o ponto de vista do usuário.

Segundo Bradshaw (1997), definir agentes com base no que eles podem fazer pode levar facilmente a um limite, onde praticamente qualquer entidade possa ser classificada como um agente. Shoham apud Bradshaw (1997, p.6) ilustra tal situação da seguinte forma: “É perfeitamente coerente tratar um interruptor de luz como um agente (muito cooperativo) com a capacidade de transmitir corrente, o qual invariavelmente transmite corrente quando acredita que nós queremos que transmita e não caso contrário; apertar o botão do interruptor é simplesmente nosso modo de comunicação de nossos desejos...”. Apesar de coerente, tal visão é desaconselhável segundo Shoham pois “...não acrescenta, desde que nós entendemos o mecanismo suficientemente para ter uma forma mais simples de descrição do seu comportamento...”.

Visando facilitar a distinção entre sistemas simples demais para serem descritos como agentes, e aqueles que realmente devem ser assim tratados, Dennett apud Bradshaw (1997, p.7) descreve três instâncias:

- Instância Física: O comportamento pode ser previsto baseado em características e/ou leis físicas.
- Instância de Projeto: O comportamento pode ser previsto baseado naquilo que a entidade foi projetada para fazer.
- Instância Intencional: O comportamento pode ser previsto baseado na admissão de um comportamento racional.

Na primeira instância, seriam classificados os sistemas naturais como, por exemplo, a colisão de bolas de bilhar, pois o comportamento pode ser explicado e previsto baseado apenas em leis físicas. Na segunda instância, estariam classificados sistemas sobre os quais existe conhecimento suficiente para prever o seu comportamento baseado no seu projeto, como, por exemplo, automóveis ou o interruptor de luz, citado nesta seção. Na terceira e última instância, seriam classificados os sistemas que não pudessem ser classificados nos dois anteriores, mas que tivessem um comportamento minimamente previsível baseado num certo grau de racionalidade. Como exemplos desta instância, poderiam ser citados animais, pessoas e agentes, pois embora seu comportamento não possa ser descrito baseados em leis físicas ou regras preestabelecidas, estes seguem raciocínios lógicos para (a maioria de) suas ações.

3.2.2. Abordagem Descritiva

Outra abordagem para se chegar a uma definição de agentes é através da descrição dos atributos que um determinado programa precisaria ostentar para poder ser classificado como tal. Franklin e Graesser (1996) citam como características comumente observáveis em agentes, as apresentadas na Tabela 3.1:

Tabela 3.1.: Características comuns em Agentes Autônomos, extraídas de Franklin e Graesser (1996).

Características	Outros nomes	Significado
Reatividade	Perceber e atuar	Responde a mudanças no ambiente.
Autonomia		Controla suas próprias ações.
Orientação a metas	Pró-ativo	Não age simplesmente em resposta ao ambiente.
Continuidade Temporal		É um processo em execução contínua.
Comunicabilidade	Habilidade Social	Comunica-se com outros agentes e possivelmente pessoas.
Aprendizado	Adaptativo	Muda seu comportamento com base em experiências anteriores.
Mobilidade		Capacidade de transportar-se de uma máquina para outra.
Flexibilidade		As ações não são pré-definidas.
Personalidade		Apresenta estados emocionais e uma “personalidade” crível.

Ainda segundo Franklin e Graesser (1996), apenas as quatro primeiras características descritas na tabela 3.1 seriam essenciais para que um sistema fosse classificado como um agente. As demais características seriam úteis para criar subclassificações para os agentes, como por exemplo: agentes móveis, agentes comunicativos e outros. De fato, várias taxonomias e esquemas de classificação foram propostos pela comunidade de pesquisa para prover um mecanismo de estruturação do espaço de agentes.

Um influente artigo de pesquisadores da IBM (Gilber apud Bradshaw 1997, pp.9) define agentes inteligentes baseado em três “dimensões” : agência, inteligência e mobilidade, como mostra a figura 3.1.

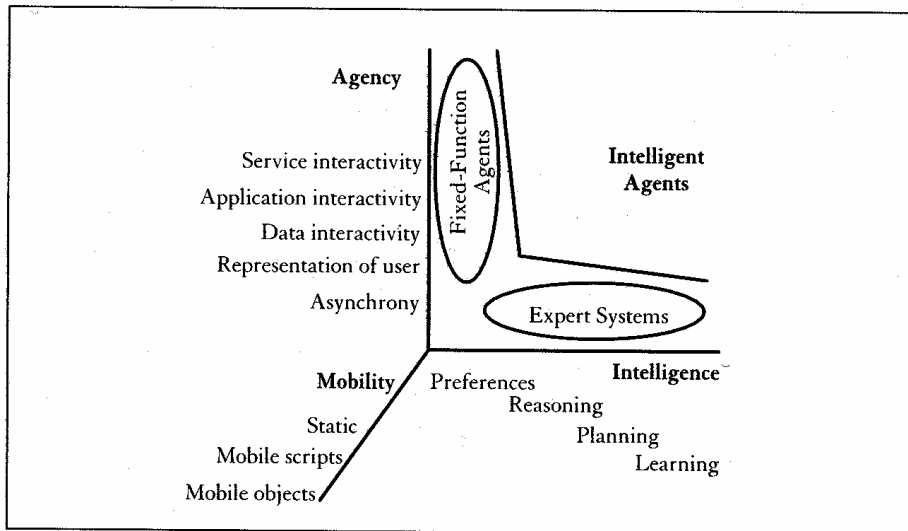


Figura 3.1. Escopo de Agentes Inteligentes, extraído de Bradshaw (1997).

A partir destas dimensões e dos diferentes graus de profundidade em cada dimensão, pode-se identificar e classificar os agentes e diferenciá-los de outros sistemas, como os sistemas especialistas. Para tanto, é importante conhecer as definições de cada uma dessas dimensões (Bradshaw, 1997):

- **Agência:** Grau de autonomia e autoridade do qual é investido o agente. Pode ser mensurado qualitativamente pela natureza das interações entre o agente e outras entidades no sistema. No mínimo, um agente deve ser capaz de trabalhar de modo assíncrono. O grau de agência é maior se o agente representa o usuário de

algum modo e/ou se pode interagir com dados, aplicativos, serviços e outros agentes.

- **Inteligência:** Grau de raciocínio e aprendizado observado em um agente. É representada pela capacidade do agente em aceitar as metas dadas pelo usuário e realizar as tarefas delegadas. No mínimo, deve existir algum grau de preferências. Níveis mais altos de inteligência incluiriam um modelo sobre o usuário, raciocínio e, em níveis ainda mais altos, aprendizado e adaptabilidade ao ambiente, em termos dos objetivos do usuário além dos recursos disponíveis para o agente.
- **Mobilidade:** Capacidade dos próprios agentes se moverem através da rede. É importante diferenciar “*Scripts*” móveis de Objetos móveis. No primeiro caso, estes podem residir em uma máquina e serem enviados a outra para a execução. No segundo caso, os Objetos móveis podem ser transportados de uma máquina para a outra no meio da execução, mantendo seus dados.

As várias tentativas feitas por pesquisadores de criar formas de distinguir entre agentes e programas comuns levaram Petrie (Petrie apud Bradshaw, 1997, p.10) a discutir tais tentativas e sumariá-las. Neste trabalho, ele demonstra que a maioria dos atuais “agentes” de busca e filtragem baseados na “*web*”, apesar de úteis, são mecanismos de resposta a consultas diretas e isoladas que seriam mais bem descritos pelo termo computacional servidor. Da mesma forma, processo móvel seria um termo mais adequado para “*applets*” Java sem funções inerentes a agentes.

Em contraposição aos trabalhos analisados por Petrie, que se caracterizavam por tentar descrever agentes em geral através de um conjunto de características não ambíguas, Petrie se

baseou em uma classe específica de agentes: agentes de mensagens estruturadas (*typed-message agents*). Os agentes de mensagens estruturadas se distinguem dos demais por se comunicarem através de um protocolo compartilhado como o KQML (Finin et al., 1994). Neste protocolo, pelo menos alguns itens da semântica da mensagem devem ser independentes da aplicação. Além disso, o protocolo de transporte não pode ser apenas cliente/servidor, mas também ser capaz de fazer comunicação ponto a ponto. Petrie define agente baseado nestes conceitos: “Um módulo de software individual não é um agente se pode comunicar-se com outros agentes candidatos apenas com um protocolo cliente/servidor sem degradação do desempenho da tarefa coletiva” (Petrie apud Bradshaw, 1997, p.11).

Franklin e Graesser (1996), estabelecem a seguinte definição para um agente autônomo:

“Um agente autônomo é um sistema situado em um ambiente sendo parte do mesmo, que o percebe e sobre o qual atua, continuamente no tempo, em perseguição de sua própria agenda e de modo a afetar o que percebe no futuro”.

Entretanto, reconhecem que através desta definição até mesmo um termostato (ou uma bactéria) poderia ser classificado como um agente autônomo. Por tal definição ser ampla demais para ser útil, propõem requisitos adicionais para diferentes proposições e, assim, gerar subclasses úteis de agentes. Entretanto, é importante notar que apesar de ser bastante ampla, a definição exclui uma série de programas tradicionais, mesmo que se tentasse forçar a definição. Por exemplo, um programa de folha de pagamento não poderia ser classificado como um agente, pois apesar de ser possível afirmar que ele percebe o ambiente através de suas entradas e o afeta através de suas saídas, a impressão da folha de pagamento num determinado mês não afeta os dados salariais do mês seguinte. Além disso, tal classe de

programas não atua continuamente no tempo sobre o ambiente, apenas é executada uma vez e depois espera ser chamada novamente.

Segundo Franklin e Graesser (1996), a maioria dos programas ordinários seria descaracterizada como agentes por uma ou ambas condições. No contexto deste trabalho, será adotada como referência a definição proposta por Franklin e Graesser (1996). Além da definição, utilizar-se-á a taxonomia de Franklin e Graesser, a qual será abordada na seção 3.3.

Finalmente, pode-se afirmar que, como muitos outros termos de computação, aquilo que começou como uma metáfora (caso de agente, “*mouse*” e outros) acaba se transformando na denominação de um artefato concreto de software. Como observa Bradshaw (1997, pp.11), “o termo (agente) irá significar o que todos entenderão, porque eles viram muitos exemplos disso ou irá cair em desuso por descrever um conceito que não é mais apropriado. O que é improvável de desaparecer são as motivações que incitaram o desenvolvimento de software baseado em agentes”.

3.3. Taxonomia de Agentes Autônomos

A taxonomia proposta por Franklin e Graesser (1996) cobre a maior parte dos exemplos encontrados na literatura, segundo seus autores. Abaixo desta classificação inicial, podem-se criar classificações através de estruturas de controle, ambientes (bancos de dados, sistemas de arquivos, rede, Internet), linguagens de programação ou ainda através da aplicação do agente. Pode-se, por exemplo, classificar as estruturas de controle por seu tipo de controle: algorítmico, baseado em regras, lógica nebulosa (*fuzzy*), redes neurais, aprendizado

computacional (*machine learning*), etc. Pode-se também utilizar certas classificações binárias tendo com base as características descritas na tabela 3.1, como: agente móvel versus não-móvel, comunicativo versus não-comunicativo, emotivo (com personalidade e estado emocional) versus não-emotivo, adaptativo versus não-adaptativo. Por exemplo, é possível classificar o agente proposto por (Gilber apud Bradshaw 1997, pp.9), “*IBM Agent*”. Este realiza algumas operações em nome de um usuário ou de outro programa, seria então classificado como um agente de tarefa específica (*task-specific agent*). No capítulo 5, serão descritos os agentes do AAS, os quais são classificados também como agentes de tarefa específica.

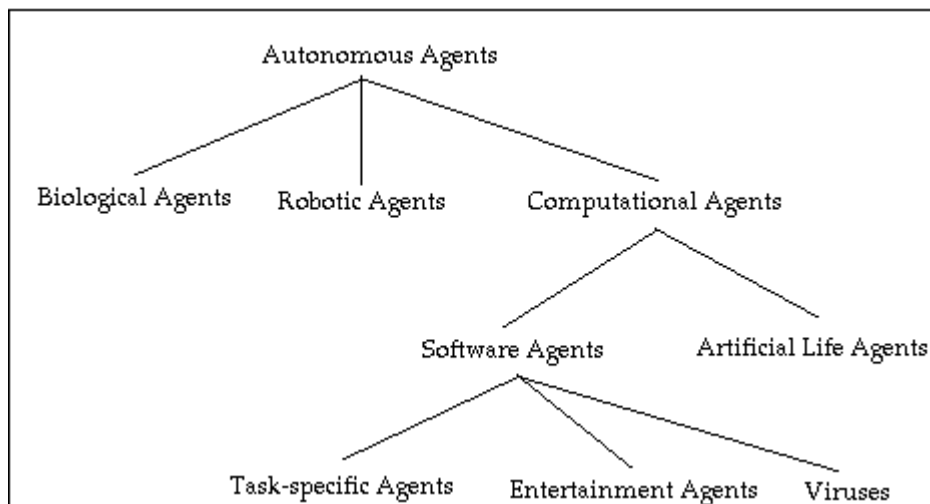


Figura 3.2. Taxonomia de Agentes por Franklin e Graesser. Extraído de Franklin e Graesser (1996)

3.4. Motivações para a construção de Agentes

Segundo Bradshaw (1997), num momento inicial os avanços na área de agentes ocorreram pelo interesse de pesquisadores nas possibilidades da inteligência artificial distribuída. Porém, fatores de ordem prática se somaram a este interesse, tais como simplificar o tratamento da complexidade inerente à computação distribuída e superar as limitações das abordagens correntes de interfaces com o usuário. Além destes, há ainda um desejo de aumentar o grau de abstração nos padrões de comunicação entre sistemas, substituindo tais padrões, baseados em interfaces programa – programa, por outros baseados em interfaces agente – agente mais poderosas, gerais e uniformes. Os dois fatores citados (a simplificação da computação distribuída e superar as limitações das atuais interfaces com o usuário) serão explorados nas seções 3.4.1. e 3.4.2.

3.4.1. Simplificando a Computação Distribuída

Um dos principais problemas em computação distribuída é a dificuldade em se estabelecer uma cooperação adequada entre os componentes de modo a otimizar o alcance das metas estabelecidas. Brodie apud Bradshaw (1997, p.12), define o termo ‘Interoperabilidade inteligente’ para tratar tal questão. Como observado por Bradshaw (1997), grande parte dos sistemas distribuídos encontrados atualmente não se comunicam ou cooperam, exceto em modos bastante simples, tais como transferência de arquivos, serviços de impressão e pesquisas em bancos de dados.

Como um pré-requisito para uma interoperabilidade inteligente, pode-se apontar a existência de padrões de comunicações independentes de plataforma. Com o advento da Internet, vários avanços foram feitos neste sentido, por exemplo os padrões HTTP, RMI e CORBA.

Na verdade, observa-se uma mudança dos sistemas operacionais de rede para uma computação baseada em redes Internet e “*intranet*” (Lewis apud Bradshaw, 1997, p.13). Esta mudança estabelecerá serviços padrões independentes de sistema operacional (como diretórios, segurança, transações, etc.), criando desta forma uma infra-estrutura para que se possa estabelecer uma interoperabilidade de mais alto nível.

Segundo Bradshaw, uma interoperabilidade de mais alto nível requererá conhecimentos da capacidade de cada sistema, bem como planejamento e alocação de tarefas, execução, monitoramento e possivelmente intervenção entre os sistemas. Para obter tais funcionalidades, pode-se utilizar um agente inteligente como gerenciador global de recursos. Entretanto, isto pode levar a criação de um gargalo e à centralização de um sistema distribuído, pois o próprio agente gerenciador de recursos seria um elo central para o sistema como um todo. A solução para tal problema seria utilizar um agente para cada sistema cooperativo, evitando deste modo a limitação de um coordenador central, conforme mostra a figura 3.3. Além disto, a comunicação entre os sistemas seria feita em nível mais alto de abstração correspondendo ao planejamento das atividades ao invés de execução das mesmas.

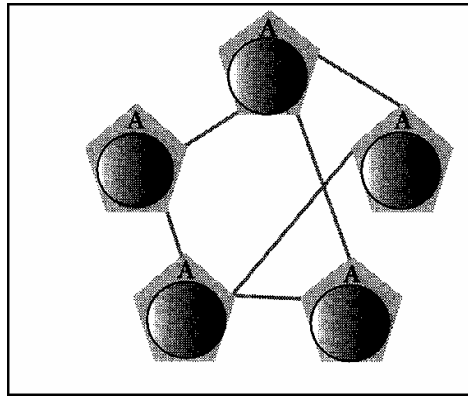


Figura 3.3. Sistemas cooperativos com agentes distribuídos. Os círculos representam os sistemas, enquanto os polígonos denotados com a letra A representam os agentes. As linhas de conexão representam comunicação em curso de agente para agente. Extraído de Bradshaw (1997).

3.4.2. Superando Limitações em Interfaces com o Usuário

A maioria das interfaces com o usuário dos programas atuais baseiam-se na manipulação direta dos objetos de software. Isto é, o usuário precisa conhecer sobre o que pode atuar a cada instante. A adoção de interfaces gráficas trouxe, sem dúvida, vantagens à medida que facilita esta visualização, especialmente quando os objetos de software têm correspondentes no mundo real ou utilizam metáforas como por exemplo arquivos, pastas, janelas, etc. Entretanto, ainda existem muitas limitações nestas interfaces. Bradshaw (1997) lista as seguintes como algumas das principais:

- **Grande Espaço de Busca:** Em grandes sistemas distribuídos pode ser difícil para o usuário encontrar o que deseja, seja navegando ou através de métodos tradicionais de indexação, especialmente, porque o número de itens pode chegar a alguns milhares.

- **Ações em Resposta a Interação com o Usuário:** Algumas vezes é interessante que, ao invés de executar uma ação imediatamente, esta seja apenas agendada para execução em horário pré-determinado ou quando ocorrer um certo evento.
- **Falta de Composição:** Não é possível compor facilmente ações básicas, objetos em ações e objetos de mais alto nível.
- **Rigidez:** A mesma consistência que faz artefatos passivos fáceis de aprender e previsíveis para tarefas simples, faz deles também elementos não confiáveis para atividades complexas.
- **Orientação à Função:** Geralmente os programas são organizados de acordo com funções genéricas de software, ao invés de organizá-los segundo uma determinada situação ou atividade do usuário.
- **Sem Capacidade de Aprendizado:** Software tradicional não percebe ou aprende com ações repetitivas de modo a se comportar mais adequadamente em relação a seu usuário no futuro.

Segundo Kay apud Bradshaw (1997, p.15), pesquisadores e desenvolvedores estão abordando tais problemas através da expressão da intenção do usuário com a noção de gerência indireta, como um estilo de interação. Nesta abordagem, os usuários ficam livres de determinar cada ação que será adotada. Ao invés disso, transmitem a um agente inteligente sua intenção e o próprio agente tomará as decisões sobre quais ações executar e em que momento fazê-lo.

Acredita-se que é possível superar as limitações apresentadas, com o uso de agentes, provendo aos sistemas baseados em agentes uma série de características, que se contrapõem aos problemas constatados de modo a eliminar (ou minimizar) as limitações existentes na abordagem de manipulação direta. Segundo Bradshaw (1997), tais características são as seguintes:

- **Escalabilidade:** agentes com capacidade de filtragem e busca podem ser executados em segundo plano (*background*) para ajudar pessoas a explorar grandes quantidades de informação.
- **Ações Agendadas ou Dirigidas a Eventos:** agentes podem ser instruídos para realizarem tarefas em tempos pré-determinados ou a reagir com uma ação específica para um evento percebido pelo próprio agente.
- **Delegação e Abstração:** Agentes podem ser feitos de modo extensível e configurável ao contrário de programas tradicionais; pois é possível se comunicar com agentes e dividir metas com os mesmos, ao invés de simplesmente enviar comandos.
- **Flexibilidade e Oportunismo:** Como agentes são instruídos em termos de metas e estratégias, podem encontrar novos caminhos para resolver problemas e explorar novas possibilidades, ajudando o usuário a resolver seus próprios problemas.
- **Orientação à Tarefa:** Agentes podem ser projetados de modo a levar em consideração o contexto de uma situação e as preferências ou tarefas do usuário.

- **Adaptatividade:** Através da identificação de padrões em ações e eventos, agentes podem utilizar algoritmos de aprendizado para continuamente melhorar seu comportamento.

3.5. Sociedades de Agentes e Sistemas Multiagentes

O poder da tecnologia de agentes reside não apenas na capacidade que um agente tem de atuar sobre o meio, mas também na capacidade de comunicar-se com outros agentes de modo a resolver problemas complexos. Este trabalho coordenado de um grupo de agentes é especialmente estudado dentro de um ramo da área de Inteligência Artificial (IA), chamado de **Inteligência Artificial Distribuída (IAD)**.

A IAD, diferentemente da IA tradicional, baseia-se na interação entre indivíduos como modelo de inteligência. Enquanto em IA se utiliza basicamente o comportamento individual como modelo de inteligência, com ênfase na representação do conhecimento e métodos de inferência, a IAD baseia-se no comportamento social como modelo de inteligência, com ênfase em ações e interações entre agentes. A IA estuda aspectos de origem psicológica, enquanto em IAD se estudam aspectos sociológicos e etológicos. A IAD divide-se em duas grandes abordagens: Resolução Distribuída de Problemas (RDP) e Sistemas Multiagentes (SMA). Em termos simples, a RDP se propõe a resolver um problema previamente definido e bem especificado através do projeto de uma sociedade de agentes; enquanto em SMA, a partir de uma coleção de agentes em uma sociedade, um determinado problema é submetido a estes agentes que o solucionam (Alvares; Sichman, 1997).

Os principais conceitos em IAD, que serão utilizados no decorrer deste texto, são descritos em seguida. Segundo Álvares e Sichman (1997), entende-se como *agente* cada entidade ativa de um sistema IAD. O *ambiente* de um sistema é o conjunto das entidades passivas visíveis aos agentes do mesmo. Um agente deve ser capaz de agir de forma autônoma, tomar decisões e agir de modo a conseguir um determinado objetivo. Desta forma, não se confunde agente com conceitos como módulos ou conjuntos de regras, requerendo um grau significativo de controle e/ou autonomia. Entende-se como *interação* a troca de informações entre agentes. Esta troca de informações pode ser feita diretamente entre os agentes, ou indiretamente através do ambiente. Um conjunto de agentes que interagem entre si e/ou com um mesmo ambiente denomina-se *sociedade*. Desta forma, pode-se afirmar que um SMA é sempre uma sociedade de agentes, embora o inverso não seja verdadeiro - nem toda sociedade de agentes é um SMA.

3.6. Conclusões

Descreve-se neste capítulo algumas definições de agentes, uma taxonomia de agentes autônomos e as principais motivações para as correntes pesquisas na área de agentes. Abordam-se ainda algumas definições da área de Inteligência Artificial Distribuída importantes para este trabalho. A definição de agente proposta por Franklin e Graesser, tal como colocada na seção 3.2, é utilizada para distinguir agentes e processos existentes no sistema proposto neste trabalho. Os agentes deste sistema são classificados segundo a taxonomia apresentada na seção 3.3. No capítulo 4, apresenta-se um conjunto de sistemas de leilão baseados em agentes autônomos.

4. SISTEMAS DE LEILÕES BASEADOS EM AGENTES

Neste capítulo, serão apresentados alguns sistemas baseados em agentes arrematantes e sistemas servidores de leilões, nos quais agentes podem atuar. Na realidade, alguns sistemas seriam classificados nos dois grupos, por serem ao mesmo tempo servidores de leilão e abrangerem agentes arrematantes, caso do eMediator (Sandholm, 2000) e do AgILE (Garcia; Lopes; Bentes, 2001). Para cada sistema serão apresentadas: suas principais características e funcionalidades, suas arquiteturas e suas limitações. Destaca-se que podem ser encontrados outros sistemas baseados em agentes com características interessantes, tais como o Tete-a-tete (Guttman; Maes; Moukas, 1998). Entretanto, os sistemas aqui apresentados foram selecionados por abordarem todas ou algumas das seguintes características:

- **Agentes arrematantes:** Agentes capazes de atuar em leilões
- **Abrangência:** Considerável abrangência de tipos básicos de leilões
- **Interface Aberta:** Possibilidade de desenvolvimento de agentes por terceiros para atuarem no sistema.

Inicialmente, o **AuctionBot** (sistema servidor de leilões), que foi desenvolvido na Universidade de Michigan (Wurman; Wellman; Walsh, 1998), será apresentado na seção 4.1. Na seção 4.2, descreve-se o **BiddingBot** (sistema de agentes arrematantes) desenvolvido na Universidade Carnegie Mellon (Ito; Fukuta; Sycara, 2000). Em seguida, na seção 4.3, mostra-se um sistema que engloba agentes arrematantes e um servidor de leilões desenvolvido no Brasil (Universidade Federal Fluminense), o **AgILE** (Garcia; Lopes; Bentes, 2001). Os sistemas BiddingBot e AgILE abordam a questão de múltiplos leilões

entretanto, como será mostrado, apresentam restrições quanto aos tipos de leilões suportados e à implementação de novas estratégias. Discute-se na seção 4.4. o sistema **eMediator** (Sandholm, 2000) desenvolvido na Universidade de Washington que incorpora agentes arrematantes e um complexo servidor de leilões. Finalmente, apresenta-se uma análise comparativa entre os quatro sistemas e um deles é selecionado como base para este trabalho na seção 4.5.

4.1.AuctionBot

O sistema AuctionBot (Wurman; Wellman; Walsh, 1998) foi desenvolvido entre 1996 e 1998, na Universidade de Michigan. Trata-se de um sistema servidor de leilões que não contempla agentes arrematantes, sendo na realidade utilizado como servidor por agentes desenvolvido por terceiros. Este sistema dá suporte não apenas os quatro tipos básicos de leilão (inglês, holandês, Vickrey e leilão de lances fechados e primeiro preço), mas também leilões duplos (*double-sided*). O AuctionBot provê interfaces para usuários humanos e agentes de software, através das quais pode-se criar e configurar leilões, dar lances e obter informações a respeito dos leilões correntes. O sistema conta ainda com um módulo para agendamento de tarefas e um banco de dados onde as informações sobre os leilões são armazenadas. A arquitetura do AuctionBot é apresentada na figura 4.1.

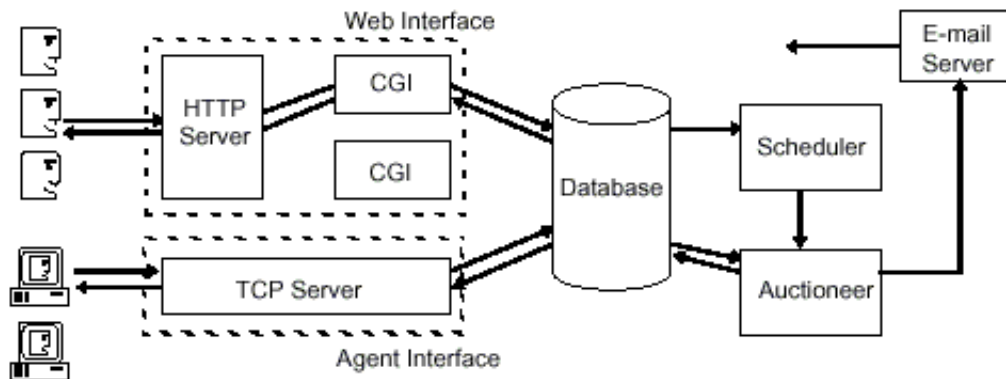


Figura 4.1. Arquitetura do AuctionBot, extraído de Wurman, Wellman e Walsh (1998). O sistema provê uma interface “Web” e uma interface para agentes baseada em conexões TCP/IP. O módulo leiloeiro (*Auctioneer*) tem capacidade de enviar mensagens eletrônicas para os usuários, este módulo é periodicamente acionado pelo módulo “*Scheduler*”.

Os agentes de software se comunicam com o sistema através de um servidor TCP. A comunicação, neste caso, é estabelecida através de conexões TCP/IP. Para usuários humanos, é disponibilizada uma interface HTML/CGI, que se comunica diretamente com o banco de dados do sistema. O sistema conta ainda com processos que realizam o trabalho do leiloeiro (*Auctioneer*). Este atualiza o estado do leilão para refletir a situação real com a chegada de novos lances e, quando for conveniente, finaliza o leilão, declarando seu vencedor e o preço estabelecido. O módulo leiloeiro pode, ainda, enviar mensagens de correio eletrônico (*e-mails*) para usuários humanos do sistema, avisando-os sobre fatos relevantes, como a chegada de novos lances ou o fechamento do leilão, por exemplo.

O AuctionBot permite um grande número de variações de leilões como citado, porém deixa o trabalho de projetar e implementar agentes para o usuário. O sistema não fornece agentes

pré-definidos que possam eliminar esse trabalho do usuário. Embora o AuctionBot possa trabalhar com vários leilões em paralelo, não é abordada a questão de desenvolver agentes que busquem melhores resultados para seus proprietários, através da participação em vários leilões simultaneamente.

4.2. BiddingBot

O BiddingBot (Ito; Fukuta; Sycara, 2000) trabalha com um grupo de agentes para monitorar e atuar em vários sistemas ou sítios de leilões distintos. Este sistema não contempla um servidor próprio de leilões, ao invés disso utiliza-se de sítios de leilões na Internet. O grupo de agentes arrematantes é controlado por um agente líder, que facilita a cooperação entre os demais e se comunica com o usuário humano. Apresenta-se na figura 4.2 a arquitetura do BiddingBot.

A atuação dos agentes do BiddingBot se dá através da coordenação feita pelo agente líder e da cooperação entre os agentes de lances (*bidder agent*). O agente líder do BiddingBot recebe mensagens dos demais solicitando autorização para efetuarem um lance. Caso nenhum agente tenha lance ativo (o lance vencedor naquele determinado instante), o agente de lances que fez a primeira solicitação, por exemplo, o agente A, recebe autorização para submeter seu lance e, após fazê-lo, informa o agente líder. Por outro lado, quando já há um agente com lance ativo, por exemplo, o agente B, o agente líder retorna ao agente solicitante o identificador do primeiro. O agente solicitante envia ao agente B uma proposta de novo lance. Este compara seu próprio lance com o lance recebido e decide se aceita a proposta. Em caso afirmativo, envia uma mensagem de aceite ao agente A e cancela o próprio lance. Caso contrário, nega a proposta de lance do agente A. Naturalmente, quando um agente não

pode cancelar seu próprio lance devido a regras do sítio com o qual opera e tem um lance ativo, este não aceita proposta de outros agentes.

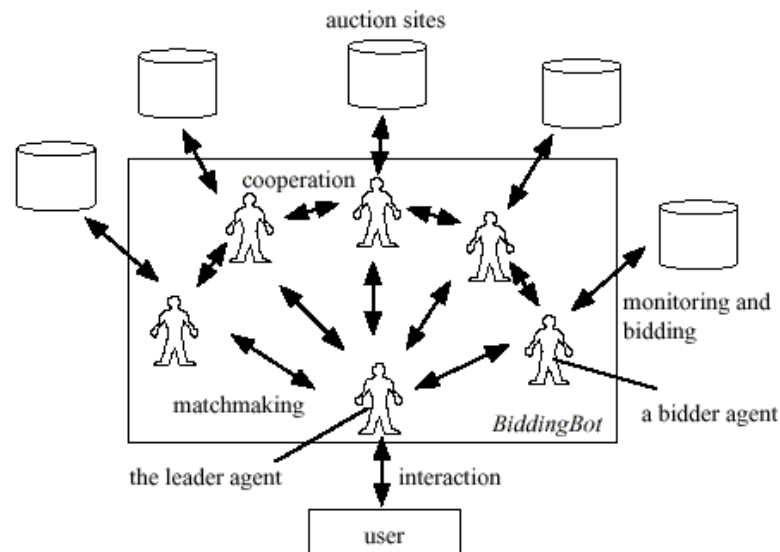


Figura 4.2. Arquitetura do BiddingBot, extraída de Ito, Fukuta e Sycara (2000). Cada agente arrematante (*bidder agent*) é associado a um sítio de leilões, que monitora e no qual, eventualmente, oferece lances sob a coordenação do agente líder (*leader agent*).

O BiddingBot trabalha com múltiplos leilões, porém todos obrigatoriamente devem ser leilões ingleses e utiliza uma estratégia de atuação pré-definida, como apresentado. Além disso, o usuário não tem como utilizar novas estratégias, sem ser obrigado a projetar e implementar todo um novo grupo de agentes.

4.3. AgILE

O AgILE (Garcia; Lopes; Bentes, 2001) é um sistema desenvolvido no Brasil e utiliza a arquitetura apresentada na figura 4.3, onde os agentes participam de leilões estabelecidos em sítios (servidores) comerciais e/ou em um servidor que faz parte do próprio AgILE.

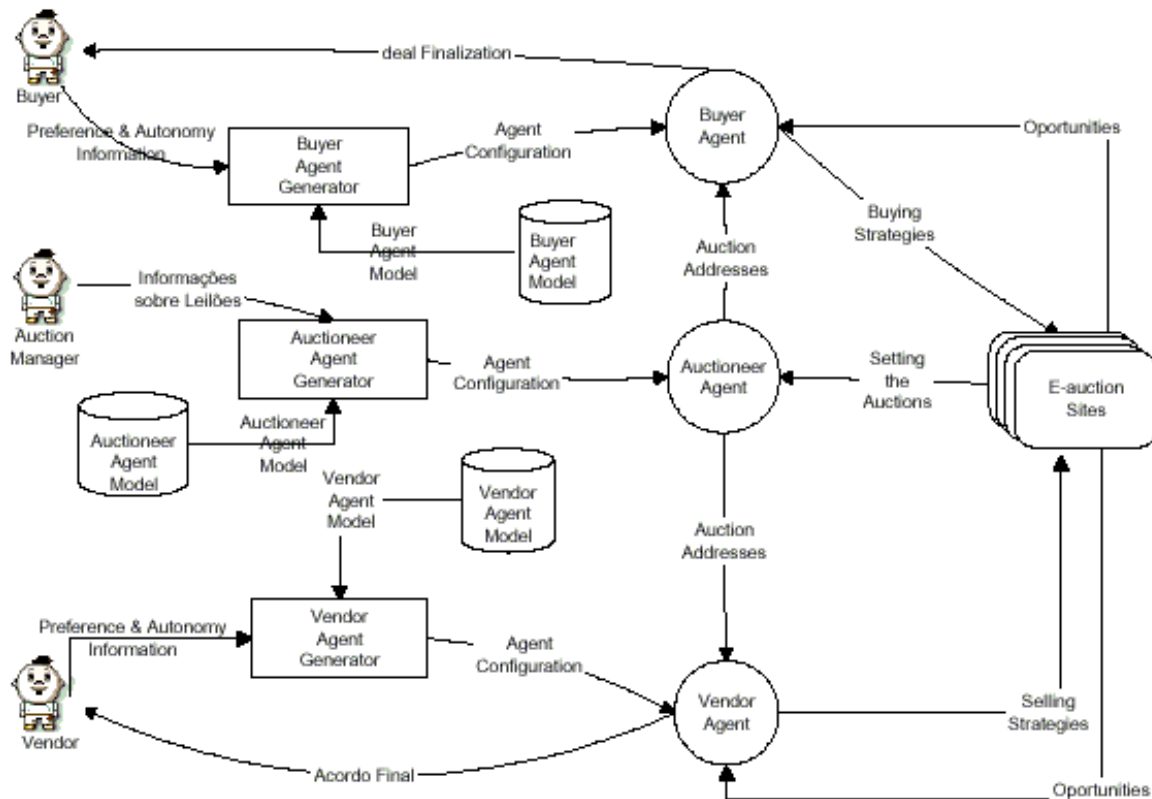


Figura 4.3. Arquitetura do AgILE, extraída de Garcia, Lopes e Bentes (2001). O sistema contempla três geradores de agentes (representados por retângulos). Os agentes podem ser de três tipos: comprador, leiloeiro e vendedor (representados por círculos) que interagem com sítios de leilões na Internet.

O sistema AgILE baseia-se em três componentes principais: Gerador de Agentes, Agentes e os Sítios de Leilão. Estes componentes são descritos a seguir:

- **Gerador de Agentes:** Cria os agentes de todos os tipos possíveis (Vendedor, comprador e leiloeiro) de acordo com os parâmetros passados pelo usuário, de acordo com a representação do agente e de seu comportamento.
- **Agentes:** Representam o usuário, seja este vendedor, comprador ou leiloeiro e para cada tipo de usuário há o respectivo agente. Atuam em vários leilões simultaneamente buscando a melhor oportunidade para fechar negócios. Seu comportamento é configurado pelo Gerador de Agentes, incluindo sua autonomia e tempo de vida.
- **Sítios de Leilão (*E-auction Sites*):** Podem ser de dois tipos sítios comerciais disponíveis na Internet e o servidor de leilões AgILE, que foi desenvolvido dentro do mesmo projeto. Os agentes atuam nos leilões mantidos por tais servidores.

O Agente Comprador (BAG) seleciona os servidores que oferecem os bens de interesse do usuário, eventualmente excluindo aqueles que não atendem às suas restrições. Baseado nas preferências do usuário, na Teoria da Decisão e também em Teoria dos Jogos, o BAG define o valor do lance e em que leilão oferecê-lo. Por outro lado, o Agente Vendedor (VAG) busca o(s) melhor(es) sítio(s) para oferecer seus produtos. Posteriormente, monitora os lances recebidos por seus produtos e decide quando sair de um determinado servidor ou quando esperar para obter um preço melhor. O agente leiloeiro, por sua vez, busca e cria leilões que irão prover os outros dois tipos de agentes (BAG e VAG). Os agentes do AgILE podem atuar em leilões do tipo inglês ou holandês, sempre seguindo o modelo tradicional, onde o

leiloeiro é um vendedor. As estratégias de atuação são definidas pelo Agente Comprador de acordo com o tipo de leilão e utiliza funções de utilidade lineares que variam à medida que se aproxima o momento do fechamento do leilão.

Os agentes do AgILE podem, ainda, operar com múltiplos leilões, porém apenas com leilões ingleses ou holandeses, conforme já citado. Além disso, como não há interface pública para agentes desenvolvidos por terceiros, o usuário fica restrito aos agentes que podem ser gerados pelo próprio sistema.

4.4. eMediator

O eMediator (Huai; Sandholm, 2000) (Sandholm, 2000) é um servidor de Comércio Eletrônico composto de um software de apoio à configuração e execução de leilões entre pessoas e/ou agentes de software, o eAuctionHouse. Contém dois outros componentes: eCommitter (Sandholm; Ferrandon, 2000) e eExchangeHouse (Sandholm; Lesser, 1996), respectivamente um componente de apoio à determinação de preço de contrato e taxas de punições em caso de rescisão, e um planejador de entrega e pagamentos de produtos que possibilita trocas entre anônimos. O presente trabalho restringe-se apenas ao eAuctionHouse, portanto não são descritos os outros dois componentes. A arquitetura da eAuctionHouse é exibida na figura 4.4 .

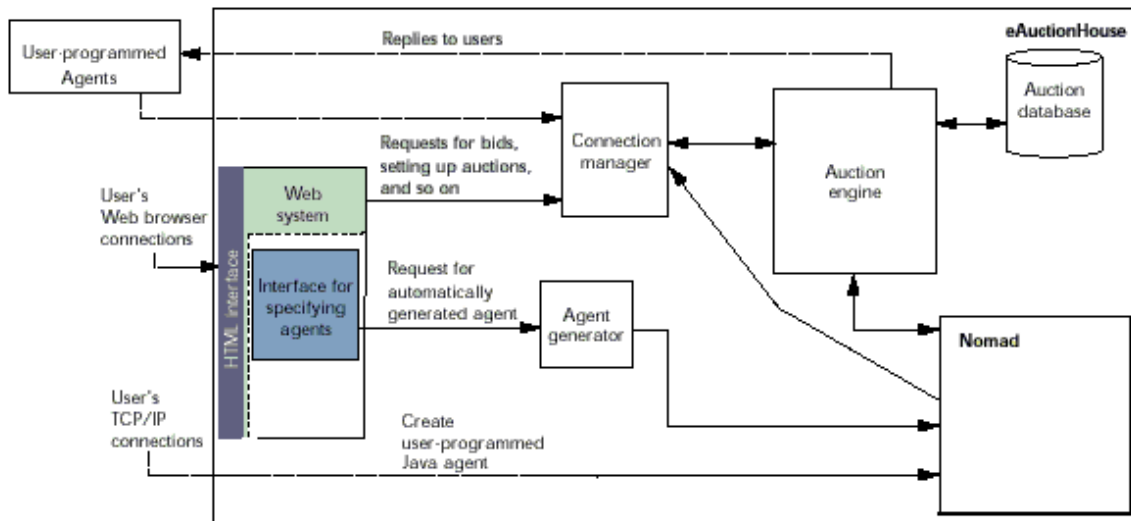


Figura 4.4. Arquitetura da eAuctionHouse, adaptado de Huai e Sandholm (2000). O sistema contempla um gerador de agentes, um módulo chamado de Nomad (responsável pelo apoio a agentes móveis) e o módulo “*Auction engine*” que realiza a configuração de leilões, recebimentos de lances, etc. No canto superior esquerdo, um retângulo representa os sistemas de agentes desenvolvidos por terceiros e que se comunicam com a eAuctionHouse através do módulo Gerente de Conexões (Connection Manager).

O eMediator contempla leilões ingleses, holandeses, Vickrey e de lances fechados e primeiro preço, além de leilões duplos (bolsas) e, também, leilões combinatórios. Os lances podem ser submetidos com valores definidos ou através de gráfico preço-quantidade. O eMediator contempla ainda agentes móveis, através de um subcomponente chamado NOMAD (Huai; Sandholm, 2000), que entretanto foge ao escopo deste trabalho. O eMediator contempla, segundo seus autores, o maior conjunto de tipos de leilões distintos dentre os servidores de leilão conhecidos. Porém, não conta com agentes que atuem em múltiplos leilões simultâneos, seja de tipos diferentes ou do mesmo tipo.

4.5. Análise Comparativa

Nesta seção, será apresentada uma breve análise comparativa entre os sistemas apresentados, considerando características de interesse do ponto de vista do estudo da atuação em múltiplos leilões simultâneos. Salienta-se que não se pretende, com esta comparação, julgar a qualidade dos sistemas apresentados, apenas sua adequação em ser utilizado como base para o trabalho proposto. A tabela 4.1 apresenta uma comparação considerando quatro características, que são descritas a seguir:

- **Agentes Arrematantes em Múltiplos Leilões:** Se o sistema dispõe de agentes capazes de atuar em múltiplos leilões simultaneamente.
- **Apoio a Agentes Desenvolvidos por Terceiros:** Se o sistema mantém uma interface pública para que agentes desenvolvidos por terceiros possam monitorar e oferecer lances nos seus leilões.
- **Tipos Básicos de Leilões Contemplados:** Relação dos tipos básicos (inglês, holandês, americano e leilão de Vickrey) contemplados pelo sistema.
- **Sistema Arrematante ou Servidor de Leilões:** O sistema se propõe a funcionar como um servidor de leilões, um sistema arrematante ou ambos.

Tabela 4.1 - Comparação entre Sistemas

Sistema	AgILE	BiddingBot	AuctionBot	eMediator	eMediator + AAS
Agentes Arrematantes em Múltiplos Leilões	Sim	Sim	Não	Não	Sim

Apoio a Agentes Desenvolvidos pelo Usuário	Não	Não	Sim	Sim	Sim
Tipos Básicos de Leilões Contemplados	Inglês, Holandês	Inglês	Todos	Todos	Todos
Sistema Arrematante ou Servidor de Leilões	Ambos	Arrematante	Servidor de Leilões	Ambos	Ambos

O eMediator destaca-se por oferecer interface pública para o desenvolvimento de agentes por terceiros e por contemplar o maior número de tipos de leilões. Além disso, foi fundamental para a escolha do eMediator como base para este trabalho, ter-se obtido a colaboração dos desenvolvedores do sistema, em especial Tuomas Sandholm e Kate Larson.

4.6. Conclusões

Neste capítulo, descreveu-se um conjunto de sistema de leilões baseados em agentes e se identificou a ausência de um sistema de leilões com agentes arrematantes com capacidade de atuar em múltiplos leilões, nos quatro tipos básicos de leilões. No capítulo 5, descreve-se o sistema AAS baseado no eMediator que supera esta deficiência.

5. PROJETO DO SISTEMA AAS

Neste capítulo, será descrito um sistema para apoiar a criação de agentes que atuem simultaneamente em múltiplos leilões, chamado de **AAS** (*Auction Agent System*), bem como sua arquitetura e funcionalidades. As características determinantes para a escolha do eMediator como base para o presente trabalho foram sua abrangência em tipos de leilão, a interface pública para o desenvolvimento de agentes por terceiros e a colaboração dos autores do sistema, como citado . Salienta-se, entretanto que o AAS pode ser facilmente estendido para operar com outros servidores de leilão em futuros trabalhos. Na seção 5.1, descreve-se as principais características dos agentes existentes no AAS. A arquitetura do sistema é detalhada na seção 5.2. Em seguida, são descritos os três agentes do AAS: o agente de busca de leilões (seção 5.3), o agente de monitoramento de leilões (seção 5.4) e o agente arrematante (seção 5.5). Finalmente, descrevem-se o funcionamento do sistema na seção 5.6, as estratégias de atuação dos agentes arremates implementados na seção 5.7 e conclui-se o capítulo na seção 5.8.

5.1. Agentes no AAS

O termo agente tem uma grande variedade de definições, com grande variação de complexidade entre autores, como explorado na seção 3.1. Neste trabalho, será seguida a definição proposta por Franklin e Graesser (1996). Utilizar-se-á ainda a taxonomia proposta por Franklin e Graesser (1996) para classificar os agentes existentes no AAS. Nesta taxonomia (Franklin; Graesser, 1996) são apresentadas algumas classificações de agente de software: agentes móveis, comunicativos, agentes de tarefa específica, etc. Dentre estas, uma classificação é de especial interesse para o AAS: agentes de tarefa específica. Como

exemplo de agente de tarefa específica, pode-se citar o “*IBM Agent*” (Franklin; Graesser, 1996), que é definido como uma entidade de software que executa um conjunto de operações em nome do usuário ou de outro programa com algum grau de independência e que emprega algum conhecimento ou representação dos desejos ou metas do usuário na execução desta tarefa.

O AAS conta com três tipos de agentes: **agente arrematante** (*Bidder*), **agente de busca de leilões** (*AuctionSearcher*) e **agente de monitoramento de leilões** (*AuctionMonitor*). Vários agentes arrematantes podem ser executados concorrentemente compartilhando os serviços dos outros agentes, conforme mostra a figura 5.1.

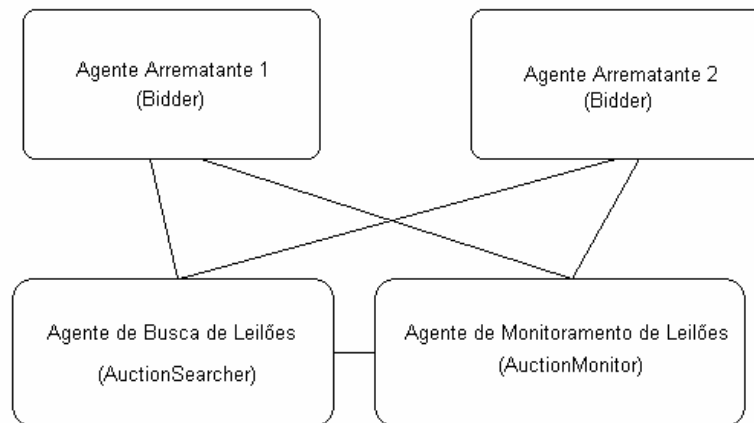


Figura 5.1. Exemplo de atuação de agentes AAS. As linhas simbolizam comunicações entre os agentes. Diferentes agentes arrematantes utilizam os serviços dos mesmos agentes de busca e monitoramento.

Todos os agentes do AAS seguem a definição de Franklin e Graesser (1996) para agentes autônomos, o que pode ser demonstrado através da obediência dos agentes às quatro primeiras características da tabela 3.1: reatividade, autonomia, orientação a metas e continuidade temporal. Segundo Franklin e Graesser, apenas estas quatro características são

essenciais para um agente (Franklin; Graesser, 1996). Na tabela 5.1, são demonstradas as características essenciais para cada agente do AAS.

Tabela 5.1. Agentes AAS e suas características essenciais

	AuctionMonitor	AuctionSearcher	Bidder
Reatividade	Responde a alterações em um leilão, enviando mensagens para o Bidder.	Responde a criação de um novo leilão de interesse para o usuário.	Responde à superação de seu lance, oferecendo novo lance (se for o caso).
Autonomia	Monitora, sem controle externo, uma série de leilões e informa quando novas informações estão disponíveis.	Busca novos leilões de interesse e, caso encontre, solicita seu monitoramento.	Decide o valor do lance e o leilão alvo .
Orientação a metas	Atua no sentido de manter informado o(s) Bidder(s) sobre os leilões de interesse do próprio.	Age para encontrar o maior número possível de leilões para o usuário.	Atua para conseguir o produto desejado pelo usuário.
Continuidade Temporal	Permanece continuamente em execução.	Permanece continuamente em execução.	Permanece continuamente em execução.

Os agentes arrematantes e os agentes AAS (AuctionSearcher e AuctionMonitor) podem ser classificados, utilizando a taxonomia de Franklin e Graesser (1996), como agentes de tarefa

específica, pois são dedicados a um propósito específico e executam uma série de ações em nome do usuário ou de outro agente (Franklin; Graesser, 1996). O agente de busca de leilões (AuctionSearcher) faz buscas para encontrar leilões de interesse em nome do(s) agente(s) arrematante(s), enquanto o agente de monitoramento de leilões (AuctionMonitor) acompanha o desenrolar de cada um dos leilões selecionados para o(s) agente(s) arrematante(s). Por sua vez, o agente arrematante tenta comprar um determinado produto em nome do usuário. Apesar de serem classificados da mesma forma, observa-se uma maior complexidade na lógica de funcionamento do agente arrematante ao se guiar através de estratégias para conseguir o melhor resultado para seu usuário. Este conjunto de agentes é definido como uma sociedade de agentes (Franklin; Graesser, 1996).

5.2. Arquitetura do AAS

O AAS soluciona dois problemas encontrados ao se desenvolver agentes para atuarem no eMediator: interface de baixo nível para comunicação com o servidor e a ausência de serviços de busca e monitoração de múltiplos leilões (Castro; Sichman, 2003). A interface ou API do eMediator é baseada em cadeias de caracteres formatadas e transmitidas através do protocolo TCP/IP. O AAS fornece um conjunto de classes Java que encapsulam tal API, oferecendo um nível mais alto de abstração e uma interface orientada a objetos.

Os serviços de busca e monitoramento no AAS são realizados respectivamente pelos agentes AuctionSearcher e AuctionMonitor. O serviço de busca consiste em encontrar os leilões de interesse do agente, segundo as informações passadas por este, nesta implementação o nome do produto. Na figura 5.2, é apresentada a arquitetura de software do AAS e sua ligação com o eMediator.

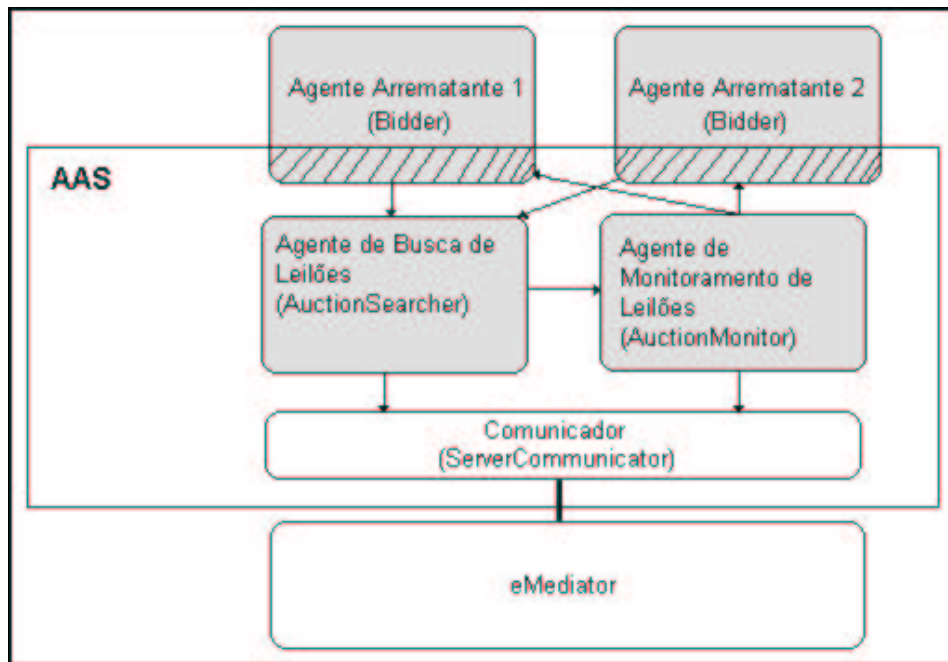


Figura 5.2. Arquitetura de Software do AAS. Os retângulos em cinza representam agentes AAS. As setas representam comunicações entre agentes e ou processo Comunicador, indicando o sentido da iniciativa da comunicação. Os agentes de busca (AuctionSearcher) e de monitoramento (AuctionMonitor) atuam para identificar os leilões de interesse dos agentes arrematantes e monitorar tais leilões, respectivamente. Estes dois agentes fazem pedidos ao processo Comunicador (ServerCommunicator), o qual faz requisições ao eMediator através da API do próprio eMediator (linha em negrito).

Do ponto de vista de implementação, o AAS conta com um processo chamado de **Comunicador** (*ServerCommunicator*). Este funciona como interface de comunicação com o servidor de leilões. Este processo não é classificado neste trabalho como um agente, pois apesar de trocar mensagens em KQML com os agentes e executar tarefas solicitadas por estes, o faz com pouca ou nenhuma autonomia. O uso de um processo em separado para

servir de interface com o servidor de leilões, ao invés de simplesmente implementar tal funcionalidade nos agentes, se justifica pela simplificação do trabalho de implementar a comunicação com novos servidores de leilões em futuras expansões, e até mesmo permitir que um agente arrematante possa participar simultaneamente de leilões localizados em servidores distintos. Para que os agentes arrematantes sejam capazes de atuar em novos servidores de leilão, seria necessário desenvolver apenas novos processos Comunicadores (um para cada novo servidor de leilões) e a estrutura de comunicação entre os outros agentes e os demais processos Comunicadores não seria alterada.

Apesar de ser um processo, o Comunicador interage com os agentes do AAS através da infra-estrutura de comunicação SACI (Hubner; Sichman, 2000). Esta abordagem foi adotada devido à flexibilidade e facilidade de utilização do sistema e para dar maior uniformidade aos mecanismos de comunicação, evitando o uso de um outro mecanismo de comunicação.

As mensagens são escritas em linguagem KQML e transportadas pelo SACI. Este sistema permite que os processos ou agentes estejam localizados em máquinas diferentes, possibilitando diversas configurações. Por exemplo, utilizando-se de máquinas exclusivas para o agente de busca de leilões, para o agente de monitoramento de leilões e para o processo Comunicador e, também, para os agentes arrematantes de cada usuário. O conteúdo das mensagens KQML trocadas dentro do sistema AAS são sempre comandos pré-definidos para tal sistema e que permitem executar todas as ações necessárias ao agente ou aos vários módulos (monitorar leilões, oferecer lances, etc.), bem como descrever lances e leilões através de seus parâmetros. A sintaxe e os principais comandos do AAS são descritos nas tabelas A.1 e A.3, respectivamente.

Desta forma, o Comunicador funciona como se fosse também um agente, sob o ponto de vista do ambiente SACI. Isto fica mais claro observando a figura 5.3, onde se demonstra o relacionamento de herança entre agentes e processos no AAS. A classe *AASProcess* herda da classe “Agent” do SACI, a qual permite o acesso aos mecanismos de comunicação e também que as classes herdadas possam ser lançadas a partir da interface gráfica do SACI (Hubner; Sichman, 2000).

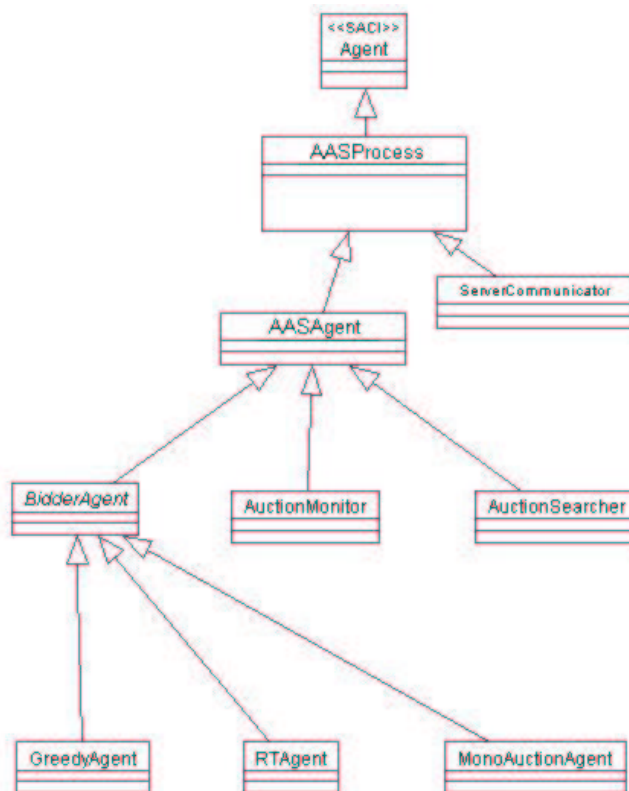


Figura 5.3. Herança de algumas classes no AAS e de alguns agentes arrematantes.

5.3. O agente de busca de leilões: AuctionSearcher

O agente de busca de leilões (AuctionSearcher) busca os leilões que estejam vendendo o produto de interesse do usuário, chamados de **Leilões de Interesse**. É importante observar que este trabalho é contínuo, pois, tipicamente, um leilão na Internet leva alguns dias ou mais e, durante este período, podem ser criados novos leilões que estejam comercializando o produto desejado. A inclusão destes novos leilões pode levar à obtenção de melhores resultados para o usuário, uma vez que aumenta o número de opções de lance para o agente arrematante. Além disto, o AuctionSearcher tem a capacidade de realizar simultaneamente buscas para vários agentes arrematantes e, por conseguinte, para vários usuários. Lista-se em seguida as principais atividades desempenhadas pelo AuctionSearcher.

- **Início:** O agente registra-se em uma sociedade definida quando do seu lançamento e publica nas páginas amarelas do SACI (Hubner; Sichman, 2000) sua habilidade de fazer busca por leilões. Além disso, procura também, neste serviço, por agentes que possam realizar a tarefa de monitoramento de leilões.
- **Registro de solicitações por agentes arrematantes:** Recebe e registra pedidos feitos por agente(s) arrematante(s). Os pedidos contêm a(s) informação(ões) necessária(s) para identificar o produto de interesse do usuário.
- **Busca e seleção de leilões:** O agente busca periodicamente no servidor de leilões, através do processo ServerCommunicator, a lista de leilões abertos. A partir desta lista e com a informação passada pelo usuário seleciona os leilões de interesse.

- **Solicitação de monitoramento dos leilões de interesse:** Uma vez identificados novos leilões de interesse, o AuctionSearcher envia mensagem ao AuctionMonitor solicitando que este passe a monitorá-los e envie as informações para o respectivo agente arrematante.

5.4. O agente de monitoramento de leilões: AuctionMonitor

A tarefa de periodicamente verificar as informações de cada leilão de interesse para o(s) usuário(s) é executada pelo AuctionMonitor. Este agente tem a capacidade de monitorar conjuntos distintos de leilões para vários agentes arrematantes, com isto podendo atender a um ou mais usuários. As principais atividades desempenhadas pelo AuctionMonitor são as seguintes:

- **Início:** O agente registra-se em uma sociedade definida quando do seu lançamento. Em seguida, publica nas páginas amarelas do SACI (Hubner; Sichman, 2000) sua habilidade de fazer monitoramento de leilões.
- **Registro de solicitações de monitoramento:** O agente registra as solicitações (vindas do AuctionSearcher) e o cliente de cada solicitação, ou seja, o agente arrematante.
- **Captção de novas informações sobre leilões:** O agente solicita, através do ServerCommunicator, informações detalhadas sobre cada leilão monitorado ao servidor de leilões.

- **Envio de informações ao(s) arrematante(s):** Ao receber novas informações de um determinado leilão, o agente verifica quais os agentes arrematantes interessados naquele leilão e lhes envia mensagens com a nova situação do leilão.

5.5. O agente arrematante: Bidder

O agente arrematante é o principal e mais complexo agente do AAS. Realiza o trabalho de definir em que leilão oferecer lances e qual o valor dos lances, de modo a obter os melhores resultados possíveis para o usuário, seguindo uma determinada estratégia. Na figura 5.4, é apresentado o diagrama de atividades para o agente arrematante (Bidder). As atividades executadas são as seguintes:

1. **Solicita Informações ao Usuário sobre o Produto e o Preço Reservado:** Capta as informações do usuário para realizar seu trabalho. De acordo com a estratégia empregada, podem ser solicitadas outras informações além do preço reservado e da identificação do produto. Por exemplo, na estratégia RT solicita-se, adicionalmente, o tempo máximo para efetuar a compra.
2. **Busca nas Páginas Amarelas por Agentes que Realizem Serviços de Busca de Leilões:** Busca no serviço de páginas amarelas do SACI (Hubner; Sichman, 2000) por agentes que realizem o serviço de busca de leilões.
3. **Solicita a um Agente com Capacidade de Busca por Leilões (AuctionSearcher) para Encontrar Leilões de Interesse:** Solicita ao agente encontrado na atividade anterior que procure por leilões que estejam comercializando o produto indicado pelo usuário na atividade 1.

4. **Busca Novas Mensagens:** Espera o recebimento de novas mensagens e, ao recebê-las, determina seu processamento.
5. **Atualiza Informações sobre Leilão de Interesse:** Atualiza informações sobre um leilão de acordo com uma nova mensagem recebida.
6. **Atualiza Informações sobre Lances:** Atualiza informações sobre um lance de acordo com uma nova mensagem recebida.
7. **Avalia Situação e Adota as Ações Devidas:** De acordo com a estratégia do agente, faz a avaliação dos leilões e dos lances e, eventualmente, decide oferecer novo lance.

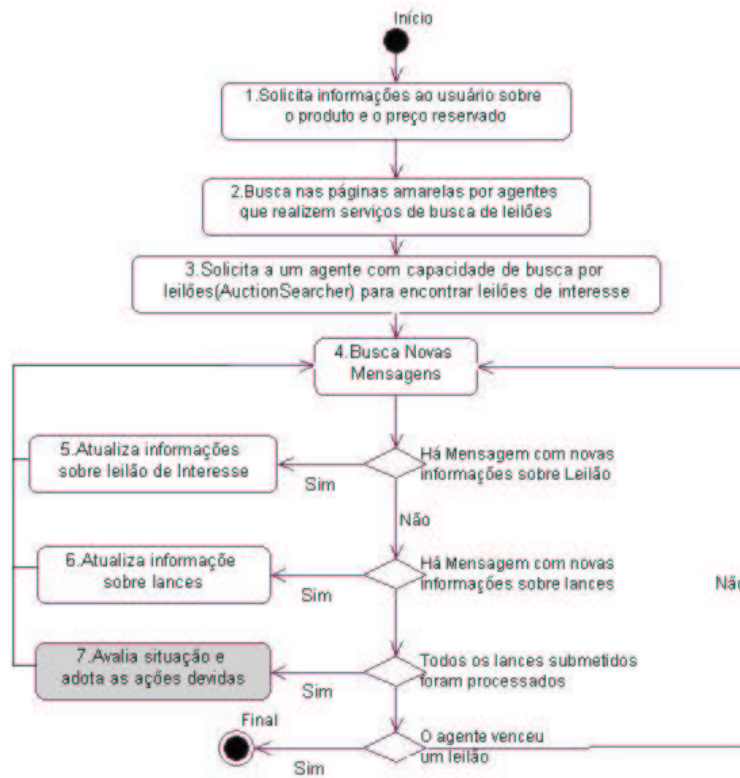


Figura 5.4. Diagrama de Atividades (Booch; Rumbaugh; Jacobson, 1999) do Agente arrematante no AAS. É importante observar que a única atividade dependente da estratégia do agente é atividade de número 7 (sete). Portanto, ao desenvolver um agente com nova estratégia, o usuário do AAS utiliza as atividades de 1 a 6, as quais já estão implementadas na superclasse de todos os agentes arrematantes no AAS, *BidderAgent*, ver figura 5.2.

Associado a cada agente arrematante do AAS, existe uma interface gráfica que é executada no mesmo processo do agente, que fornece ao usuário informações sobre os agentes que estão sendo monitorados, os lances dados pelo agente e as mensagens trocadas por este. Na figura 5.5, é apresentada um exemplo desta interface.

Status: Agent is analyzing 3 Auction(s).

Monitored Auctions

Id	Current Value	Winner Bid
289	203.0	No
290	205.0	No
291	203.0	No

Agent's Bids

Auction Id	Current Value	Active Bid	Bid Id
289	203.0	Yes	854
290	203.0	Yes	856
291	203.0	Yes	859

Received Message

Sen...	Command
Auc...	"status=Ok&msg=included.&in_response=search_auction"
Ser...	"bid_iden=&auc_iden=&bid_name=&bid_date=&status=Ok&in_response=list_bid"

Quit Refresh

Figura 5.5. Apresentação do desempenho de um agente arrematante para o usuário. No canto superior esquerdo da janela, uma mensagem é apresentada com a descrição do estado atual do agente. A primeira tabela (de cima para baixo) apresenta uma lista dos leilões monitorados com seus identificadores, valores correntes e se o agente detém o lance vencedor naquele leilão. Na segunda tabela, são apresentados os lances oferecidos pelo agente com as seguintes informações: o identificador do leilão a que se refere o lance, seu valor, se o lance ainda é válido e o identificador do lance. A terceira tabela apresenta uma lista das mensagens recebidas pelo agente.

5.6. Funcionamento do AAS

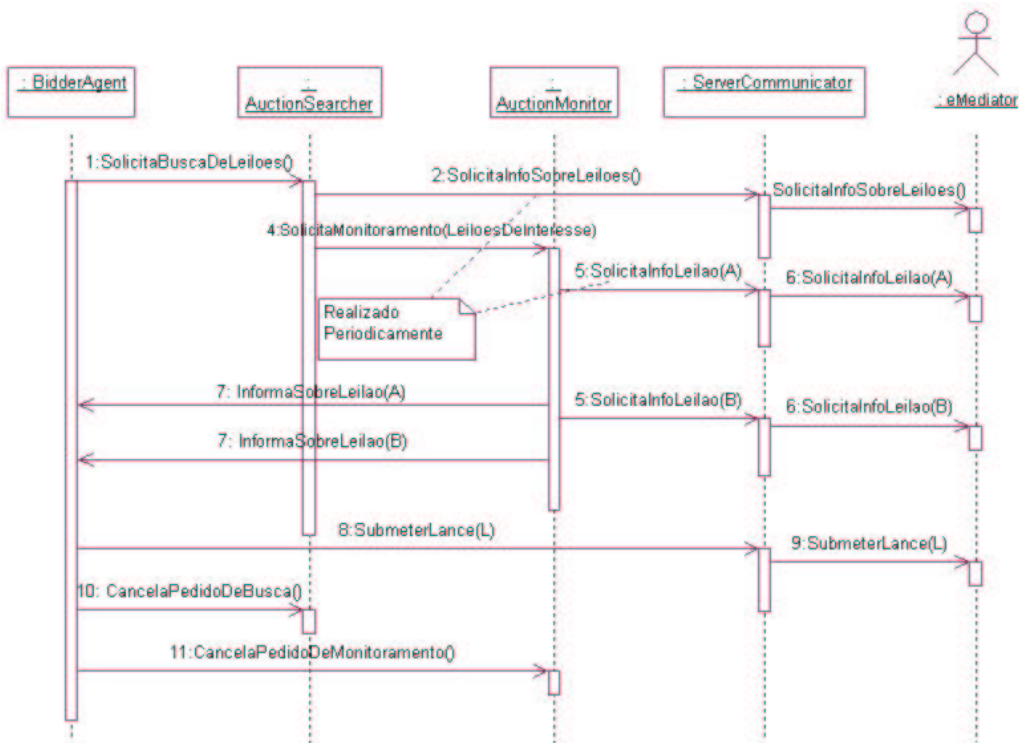


Figura 5.6. Funcionamento do AAS. Apresenta-se aqui um exemplo de execução do AAS com um agente arrematante (BidderAgent) que teve selecionados pelo AuctionSearcher alguns leilões. Estes leilões são passados ao AuctionMonitor e solicitado seu monitoramento. Após receber informações do AuctionMonitor sobre os mesmos, o agente arrematante decide oferecer lance em um leilão.

O agente arrematante (BidderAgent), bem como os agentes AuctionSearcher, AuctionMonitor e o processo ServerCommunicator são criados previamente pelo usuário, utilizando-se da interface gráfica disponibilizada pelo sistema SACI (Hubner; Sichman, 2000). Um usuário pode criar mais de um agente arrematante, mas seria desnecessário criar mais de uma instância dos demais agentes.

Uma vez criado o agente arrematante e passados a este os parâmetros necessários (como identificação do produto desejado, preço reservado e outros de acordo com a estratégia utilizada pelo agente, conforme descrito na seção 5.8) ocorrem os seguintes passos (figura 5.6):

1. O Agente arrematante (BidderAgent) solicita ao agente AuctionSearcher a identificação dos leilões de interesse através de uma mensagem AAS.
2. O AuctionSearcher envia mensagem ao ServerCommunicator solicitando informações sobre os leilões ainda abertos a lances.
3. O ServerCommunicator por sua vez solicita a lista de leilões abertos ao servidor de leilões eMediator, utilizando para tal comunicação a API daquele sistema. Ao receber tal lista de leilões, o ServerCommunicator a envia como resposta para o AuctionSearcher.
4. De posse da lista de leilões abertos, o AuctionSearcher seleciona aqueles que são de interesse do agente arrematante. Em seguida, envia mensagens ao agente AuctionMonitor solicitando o monitoramento de cada um dos leilões de interesse, neste exemplo apenas dois (leilões A e B). Periodicamente, o AuctionSearcher repete os passos de 2 a 4 para verificar se novos leilões de interesse são incluídos no servidor.
5. Após receber a lista de leilões que devem ser monitorados oriundas do AuctionSearcher, o AuctionMonitor solicita periodicamente ao ServerCommunicator informações detalhadas sobre cada um dos leilões monitorados, neste caso os leilões A e B.

6. O ServerCommunicator solicita informações ao eMediator para cada leilão determinado pelo AuctionMonitor e retorna tal informação para aquele agente.
7. De posse das informações atualizadas sobre os leilões monitorados, o AuctionMonitor informa ao agente arrematante sobre a situação corrente de cada leilão (lances submetidos, tempo de fechamento, etc.).
8. O agente arrematante decide, então, qual atitude tomar: dar lance em um determinado leilão ou aguardar. Neste exemplo, o agente decide apresentar lance em um dos leilões. Portanto, envia uma mensagem com o dados de seu lance (identificação do leilão e valor) para o ServerCommunicator.
9. O ServerCommunicator envia o lance para o eMediator através de sua API.
10. e 11. O processo de monitoração pelo AuctionMonitor continua. Como neste exemplo, logo após a chegada do lance o leilão é fechado e A é declarado vencedor. O AuctionMonitor vai identificar tal situação e informá-la ao agente arrematante. Este envia mensagens ao AuctionSearcher e ao AuctionMonitor informando que já comprou o produto, finalizando assim todo o processo.

5.7. Estratégias de Atuação dos Agentes no AAS

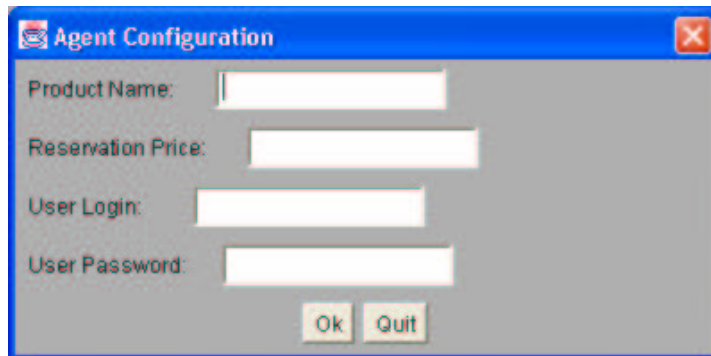
Para demonstrar a utilização do AAS, apresentam-se implementados dois agentes com estratégias de atuação distintas e que foram selecionadas de artigos sobre a atuação em múltiplos leilões. Optou-se por implementar duas estratégias para possibilitar a comparação entre os resultados obtidos por cada uma e demonstrar a capacidade do AAS de auxiliar o desenvolvimento de estratégias mais eficientes. Os agentes são: GreedyAgent e RTAgent,

que utilizam as estratégias Greedy e “*Remaining Time*” (RT), respectivamente. Estas estratégias são descritas a seguir.

- **Greedy** (Byde, 2001): Trata-se de uma das mais simples estratégias para aplicação em múltiplos leilões, onde o agente dá lance no leilão com o menor valor corrente (e aleatoriamente em caso de empate) e com preço apenas ligeiramente acima do preço corrente. O agente continua dando lances até vencer ou ser alcançado seu preço reservado, porém mantendo sempre apenas um lance vencedor no conjunto de leilões. Para atuar, o agente Greedy precisa além da informação básica de nome do produto, apenas do preço reservado do usuário para aquele item, conforme mostra a figura 5.7.
- **Remaining Time (RT)** (Anthony et al., 2001): Esta estratégia é definida com base no tempo restante para o final dos leilões. À medida que se aproxima o final do leilão, aumenta-se o valor do lance. A escolha do leilão no qual se vai oferecer lance é feita através do cálculo das utilidades esperadas em cada leilão. Caso se decida que um lance deve ser oferecido em algum leilão este será enviado ao que possuir maior utilidade esperada. Um agente pode decidir não enviar lance algum em duas situações: quando o preço reservado foi atingido ou quando se deseja evitar o comprometimento precoce, já que não existem leilões com fechamento iminente. O agente RT necessita da informação de preço reservado e do tempo máximo para compra do bem, conforme mostra a figura 5.8.

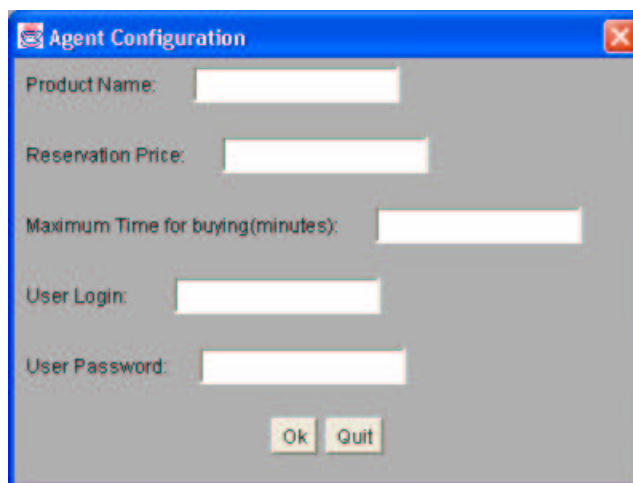
Estas estratégias foram selecionadas dentre as várias existentes pela simplicidade de implementação de ambas. A estratégia Greedy é uma generalização para o caso de múltiplos leilões da estratégia dominante para leilões ingleses. O agente apresenta o lance de menor valor possível e vai aumentando o valor do lance, até que alcance seu preço reservado. Por

outro lado, a estratégia Remaining Time apresenta, além de sua simplicidade, a capacidade de ser utilizada em qualquer um dos quatro tipos básicos de leilões.



The screenshot shows a dialog box titled "Agent Configuration" with a blue title bar and a close button (X) in the top right corner. The dialog has a grey background and contains four text input fields: "Product Name:", "Reservation Price:", "User Login:", and "User Password:". Below the input fields are two buttons: "Ok" and "Quit".

Figura 5.7. Tela de configuração do agente Greedy. Solicita-se o nome do produto a ser comprado, o preço reservado para o mesmo e as informações de “login” e senha do usuário no servidor de leilões, uma vez que o agente atuará em seu nome.



The screenshot shows a dialog box titled "Agent Configuration" with a blue title bar and a close button (X) in the top right corner. The dialog has a grey background and contains five text input fields: "Product Name:", "Reservation Price:", "Maximum Time for buying(minutes):", "User Login:", and "User Password:". Below the input fields are two buttons: "Ok" and "Quit".

Figura 5.8. Tela de configuração do agente Remaining Time (RT). Solicita-se o nome do produto a ser comprado, o preço reservado para o mesmo, e o tempo máximo para realizar a compra (em minutos), além das informações de “login” e senha do usuário no servidor de leilões.

5.8. Conclusões

Neste capítulo, descreve-se o sistema AAS, seus agentes, arquitetura de software e mecanismo de comunicação. Exibem-se, adicionalmente, as estratégias de atuação em múltiplos leilões simultâneos que foram implementadas em agentes arrematantes. No capítulo 6, discute-se alguns experimentos realizados utilizando o sistema AAS, bem como os agentes arrematantes que implementam as estratégias de atuação descritas aqui.

6. EXPERIMENTOS E AVALIAÇÃO DE RESULTADOS

Como já explicitado no capítulo 1, não se pretende neste trabalho definir estratégias ótimas para atuar em leilões múltiplos, uma vez que a complexidade de tal tarefa fugiria ao escopo de um trabalho de Mestrado. Como pode ser comprovado através do considerável número de artigos relacionados ao tema, existem enormes dificuldades de determinar uma estratégia ótima para leilões múltiplos (Anthony et al., 2001) (Byde; Preist; Jennings, 2002) (Preist; Bartolini; Philips, 2001). Analogamente, não se pretende definir qual a melhor estratégia entre as estudadas (Greedy e RT), mas demonstrar que o AAS pode servir de base para a implementação destas estratégias, reduzindo o esforço do projetista de agentes arrematantes.

Nas seções 6.1, 6.2 e 6.3, são apresentados e discutidos os experimentos realizados e seus respectivos resultados. Os resultados obtidos são avaliados na seção 6.4. Na realização destes experimentos, utilizou-se o servidor de leilões eMediator (Sandholm, 2000), o AAS e os agentes arrematantes Greedy e RT (*Remaining Time*). Além destes agentes, foi utilizado ainda um agente que implementa a estratégia dominante para leilão inglês (ver seção 2.6 para maiores detalhes), chamado de **MonoAuctionAgent**, tal agente pode ser visto também como uma simplificação da estratégia Greedy, para o caso de um único leilão. Neste trabalho, este agente é utilizado para simular a atuação de arrematantes que trabalham em apenas um leilão, uma vez que tal situação é a mais comum.

6.1. Atuação do Agente Greedy

Neste experimento, três leilões (305, 306 e 307) são disputados simultaneamente por quatro agentes do tipo Greedy. Adicionalmente, para simular a atuação de arrematantes que operam em um único leilão foram usados cinco agentes MonoAuctionAgent, sendo dois agentes no leilão 305, outros dois no leilão 306 e um agente no leilão 307. Os resultados deste experimento são mostrados na figura 6.1., onde se representa no eixo das abscissas o tempo e no eixo das ordenadas a evolução dos preços correntes.

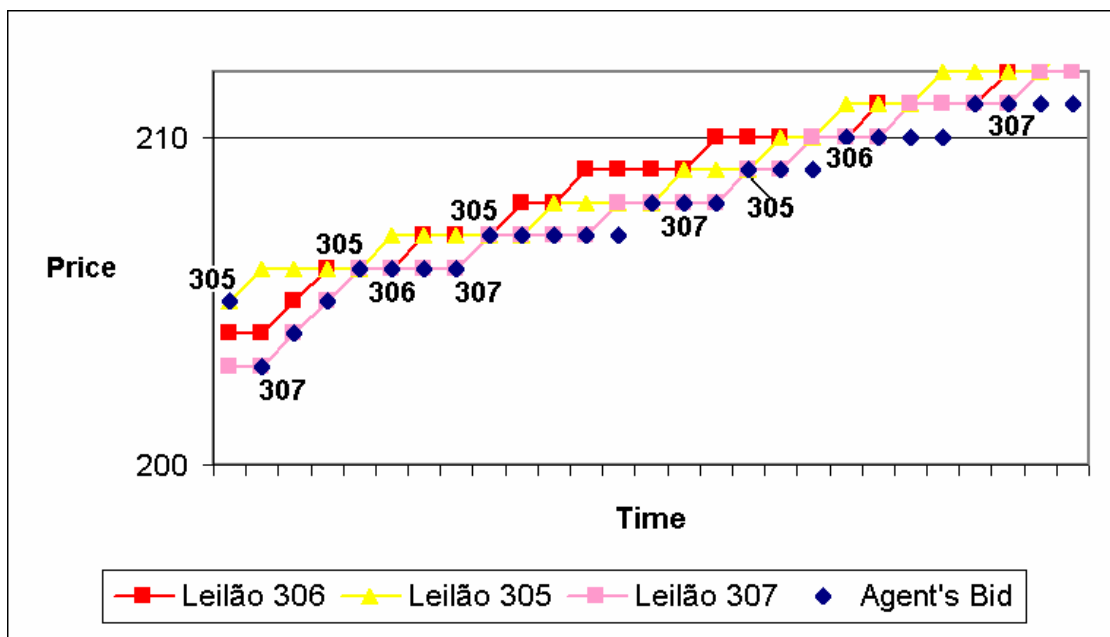


Figura 6.1. Atuação do Agente Greedy - O gráfico mostra a evolução no tempo dos preços correntes de três leilões (linhas vermelha, amarela e cinza) monitorados pelos agentes. Os lances oferecidos pelo agente Greedy são representados por um losango azul, respectivamente, junto a um lance é apresentado o número do leilão ao qual se refere.

Pode-se observar no primeiro ponto do gráfico do agente, que este estava oferecendo lance no leilão com valor mais alto (leilão 306). Isto ocorreu porque o agente não tinha conhecimento, quando da tomada de decisão, dos outros dois leilões (305 e 307). Por exemplo, os leilões podem ter sido criados após o agente ter oferecido o lance. No AAS, a busca por novos leilões é constante e feita pelo agente de busca de leilões (AuctionSearcher), conforme explicação no capítulo 5. Entretanto, apesar de estar em um leilão de preço mais alto, ao ter seu lance superado e conseqüentemente estar livre da obrigação com aquele leilão (segundo ponto), o agente passa para o leilão de preço mais baixo (leilão 307, linha cinza) e deste ponto em diante continua escolhendo o leilão de preço mais baixo entre todos os considerados. Observa-se que a linha azul, que representa o valor dos lances do agente, acompanha a evolução dos preços pelo valor mais baixo possível, não necessariamente no mesmo leilão. Isto demonstra que atuar em múltiplos leilões pode trazer economia considerável para o usuário. O possível comprador que tivesse escolhido o leilão 306 a priori pagaria mais para obter o mesmo bem, pois este leilão teve um preço final mais alto que os demais. Maiores detalhes de como foi feito o experimento, encontram-se no anexo B.

6.2. Atuação do Agente RT

Neste experimento, três leilões (326, 327 e 328) são disputados simultaneamente por dois agentes do tipo RT. Para simular a atuação de arrematantes que operam em um único leilão foram usados 6 agentes MonoAuctionAgent (2 em cada leilão).

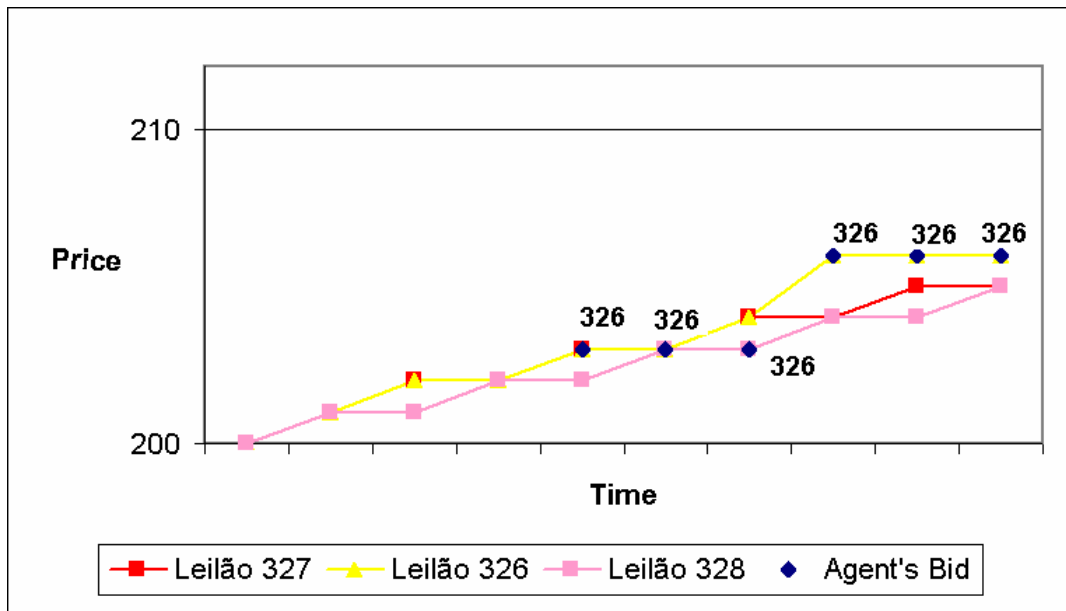


Figura 6.2. Atuação do Agente RT - O gráfico mostra a evolução dos preços correntes de três leilões (linhas vermelha, amarela e cinza) monitorados pelos agentes. Os lances oferecidos pelo agente RT são representados por um losango azul, respectivamente, junto a um lance é apresentado o número do leilão ao qual se refere.

Ao contrário do agente Greedy, o RT não se baseia nos valores correntes dos leilões para definir o valor de seu lance. Ao invés disso, baseia-se nos tempos para fechamento dos leilões, no tempo máximo especificado pelo usuário para comprar o item e no preço reservado definido por este para definir o valor de seu lance, através da fórmula (1) (Anthony et al., 2001):

$$f_{rt} = \alpha_{rt}(t) * P_r, \text{ onde } \alpha_{rt}(t) = k + (1 - k) (t_{corrente} / t_{máximo})^{1/\beta} \quad (1)$$

O preço reservado é representado na fórmula (1) por P_r , enquanto k e β são constantes definidas pelo projetista do agente. Quando $\beta < 1$, o agente mantém o valor de seu lance baixo até estar bem próximo do tempo máximo ($t_{\text{máximo}}$), neste momento é sugerido como valor do lance o próprio preço reservado. Por outro lado, quando $\beta > 1$ o agente inicia seus lances com um valor próximo ao preço reservado (Anthony et al., 2001). A escolha do leilão no qual será oferecido o lance também é feito de modo bem diverso daquele utilizado na estratégia Greedy. Faz-se uma seleção dos leilões potenciais incluindo aqueles próximos do fechamento e cujo valor mínimo (melhor lance corrente mais o incremento mínimo), seja menor que o valor do lance calculado pelo agente (f_{rt}). A escolha do leilão alvo (aquele que receberá o lance) é feita calculando a utilidade esperada para cada leilão potencial para o valor de lance definido (f_{rt}). O leilão com maior utilidade esperada será escolhido e o lance será submetido. Segundo Anthony et al. (2001), “...utilidade esperada, por definição, é o produto da probabilidade do agente vencer o leilão a um valor de lance dado e o valor da função de utilidade do agente”. A função de utilidade do agente é apresentada na fórmula (2), onde v é o valor do lance definido pela fórmula (1).

$$U(v) = 1 - (v / P_r)^{1/\beta} \quad (2)$$

Por outro lado, a probabilidade de vencer o leilão é fortemente dependente do tipo de leilão. Segundo Anthony et al. (2001), nos leilões ingleses, americanos e de Vickrey, quanto mais próximo o valor do lance estiver do preço reservado, maior a probabilidade de vitória. Em leilões holandeses, a probabilidade de vencer com lance igual ou menor que o preço reservado é próxima a 100%, devido à característica descendente dos lances neste tipo de leilão.

6.3. Comparação de Atuação dos Agentes Greedy e RT

Neste experimento, três leilões (330, 331 e 332) são disputados simultaneamente por dois agentes dos tipos Greedy e RT (um de cada tipo). Para simular a atuação de arrematantes que operam em um único leilão foram usados três agentes MonoAuctionAgent (um em cada leilão).

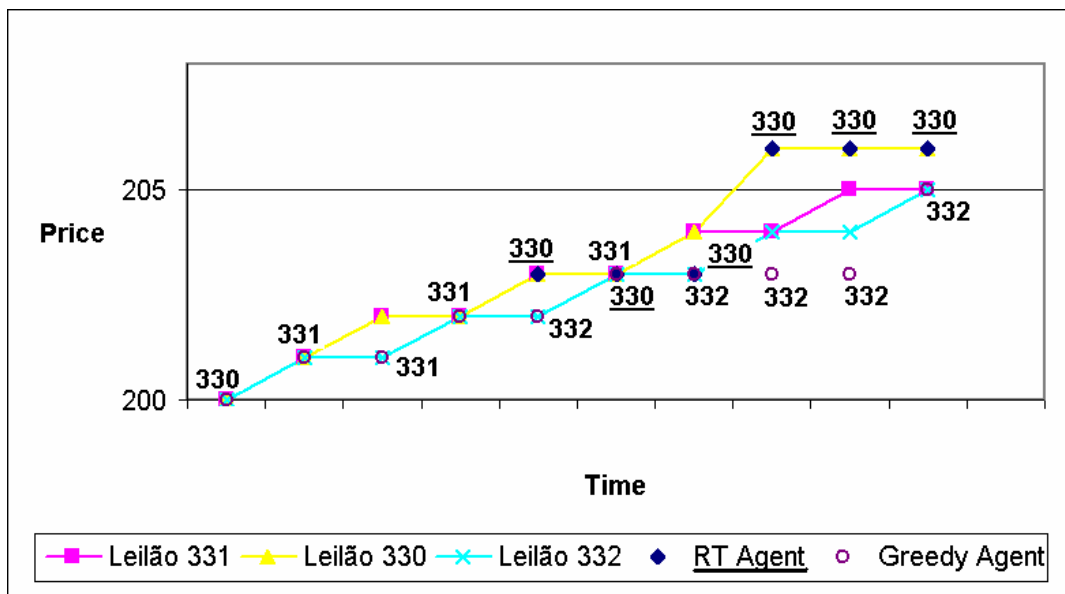


Figura 6.3. Comparação de Atuação dos Agentes Greedy e RT - O gráfico mostra a evolução dos preços correntes de três leilões (linhas vermelha, amarela e cinza) monitorados pelos agentes. Os lances oferecidos pelos agentes Greedy e RT são representados por um círculo e um losango azul, respectivamente, junto ao lance é apresentado o número do leilão ao qual se refere. No caso do agente RT, os números estão sublinhados.

O agente RT apresenta comportamento similar ao apresentado na seção 6.2, como é de se esperar devido à baixa sensibilidade da estratégia utilizada em relação à concorrência nos leilões, seja em termos qualitativos ou quantitativos, pois a grandeza que norteia as ações do agente é sempre tempo: tempo para fechamento do leilão e tempo máximo fornecido pelo usuário. Em razão das grandes diferenças conceituais entre as duas estratégias, o comportamento dos agentes também é bastante diverso. Enquanto, o agente Greedy mantém-se no leilão com valor corrente mais baixo e apresenta lances freqüentemente, o agente RT apresenta poucos lances (neste experimento, apenas dois) e com valores não diretamente relacionados ao valor do lance, podendo inclusive ser sensivelmente maior que os valores correntes dos leilões. Isto ocorre quando o tempo definido pelo proprietário para que o agente efetue a compra está próximo do fim, nesta situação o agente oferece lance com valor próximo ao preço reservado, de acordo com a fórmula (1).

Observando o gráfico na figura 6.3, pode-se chegar à conclusão que o agente RT está oferecendo lances desnecessariamente altos. Entretanto, a estratégia RT parte do princípio que o preço reservado é definido com base no preço de mercado estimado para o item em leilão. Assim, o fato de dar um lance próximo ao preço reservado não indica necessariamente que o agente pagará um preço mais alto, pois a evolução dos valores correntes continua até seu fechamento e provavelmente o preço final será próximo ao valor de mercado. Por outro lado, o agente RT praticamente garante a vitória em um leilão dentro do tempo estipulado pelo proprietário. Além disso, pode ser utilizado em leilões Vickrey, americano e holandês. Já o agente Greedy só é utilizável em leilões ingleses. Ainda em relação ao agente Greedy, este apresenta comportamento similar ao apresentado no experimento descrito na seção 6.1, exceto pelo fato de passar a não mais oferecer lances no

leilão alvo do agente RT, o que é facilmente explicável devido aos valores mais elevados oferecidos por este agente (RT).

6.4. Avaliação dos Resultados

Como citado na seção 1.1., não se pretendia neste trabalho definir estratégias ótimas para atuar em leilões múltiplos ou determinar qual a melhor dentre as estratégias implementadas, mas desenvolver um sistema para simplificar a construção de agentes para operarem no eMediator em múltiplos leilões. Tal objetivo foi alcançado, pois os experimentos apresentados neste capítulo demonstram o funcionamento de agentes arrematantes AAS que implementam duas estratégias distintas (Greedy e RT). Demonstram também o funcionamento dos agentes de busca (AuctionSearcher) e monitoramento (AuctionMonitor), partes integrantes do AAS e responsáveis respectivamente pela busca e monitoramento de leilões, e da interface de comunicação com o servidor de leilões eMediator. Os resultados apresentados nas seções 6.1 a 6.3, comprovam ainda a utilidade dos registros criados pelo AAS para avaliar as estratégias implementadas.

Por outro lado, observando o pequeno tamanho de código necessário para implementar os agentes Greedy e RT (114 e 268 linhas de código Java, respectivamente) e o fato destes não terem de controlar a busca e o monitoramento dos leilões (feito por agentes fornecidos pelo AAS) pode-se concluir que o AAS cumpre o papel de facilitar o desenvolvimento de agentes arrematantes capazes de atuar em múltiplos leilões simultâneos. Adicionalmente, foi desenvolvido ainda um agente (MonoAuctionAgent) que implementa a estratégia dominante para um leilão inglês, que facilita os experimentos de comparação entre agentes atuantes em múltiplos leilões e agentes atuantes em um único leilão.

Apesar de não ser objetivo deste trabalho, foi feita uma pequena análise comparativa entre as estratégias implementadas e pode-se perceber que a estratégia RT se mostra mais adequada aos usuários com fortes restrições de tempo, enquanto a estratégia Greedy se mostra adequada aos usuários preocupados em obter o item ao menor preço possível e que possam se limitar apenas a leilões do tipo inglês.

6.5. Conclusões

Experimentos envolvendo os dois agentes com capacidade de atuação em múltiplos leilões e um agente arrematante capaz de atuar em apenas um leilão foram apresentados neste capítulo e os resultados obtidos foram discutidos. Apresentam-se algumas considerações adicionais sobre os resultados obtidos e sugestões para trabalhos futuros no capítulo 7.

7. CONCLUSÕES

Buscou-se, neste trabalho, desenvolver uma infra-estrutura à construção de agentes para operarem no eMediator em múltiplos leilões simultaneamente. Com o intuito de facilitar o desenvolvimento de novas estratégias e a comparação dos resultados obtidos pelos agentes. Foram desenvolvidos dois agentes para múltiplos leilões, cada um deles com uma estratégia de operação diferente, para validar e comprovar tais funcionalidades. Através da avaliação de um conjunto de experimentos, utilizando-se destes dois agentes, foram levantados alguns dados que possibilitaram chegar a uma série de conclusões, as quais são explanadas na seção 7.1. Na seção 7.2, é feita uma avaliação global do trabalho e da consecução dos objetivos iniciais. Finalmente, apresentam-se algumas sugestões para trabalhos futuros e conclusões finais respectivamente nas seções 7.3 e 7.4.

7.1. Resultados Obtidos

Os resultados obtidos nos experimentos realizados envolvendo os agentes arrematantes Greedy, RT, além do agente que atua em um único leilão (MonoAuctionAgent), que foram apresentados no capítulo 6, comprovam a idéia inicial de que é vantajoso atuar em múltiplos leilões simultâneos, pois os agentes com esta capacidade conseguiram escolher o melhor leilão e assim obter o item a menor preço e/ou em menor tempo. Entretanto, o objetivo principal deste trabalho, conforme detalhado na seção 1.1, era facilitar o desenvolvimento de agentes arrematantes em múltiplos leilões através de um sistema, chamado AAS.

Demonstrar que uma atividade é mais fácil que outra nem sempre é uma tarefa simples. Acredita-se que o trabalho de desenvolver agentes arrematantes para múltiplos leilões foi

simplificado através do uso do AAS, porque uma série de atividades que teriam de ser implementadas pelo projetista do agente já se encontram disponíveis no sistema, como: a busca de leilões de acordo com a especificação do usuário, o monitoramento dos leilões encontrados, a interface com o servidor de leilões (eMediator) e a apresentação das ações do agente para o usuário. Assim, o projetista do agente arrematante fica livre de implementar tais funcionalidades e pode se dedicar integralmente a criar estratégias de atuação. Outro caminho para visualizar a diminuição de trabalho com o uso do AAS é observar o pequeno tamanho de código necessário para implementar os agentes Greedy e RT (114 e 268 linhas de código Java, respectivamente) em comparação com o tamanho dos agentes AuctionMonitor e AuctionSearcher (309 e 442 linhas de código Java, respectivamente). Adicionalmente, pode-se observar que o projetista de um novo agente terá que escrever uma única classe que herdará várias funcionalidades da classe BidderAgent, enquanto o AAS conta no total com 38 classes, distribuídas entre os agentes de busca e monitoramento, o serviço de interface com o eMediator e outras classes auxiliares. Estes fatos reforçam a tese que o AAS efetivamente poupa trabalho para o projetista de agentes arrematantes.

Adicionalmente, o AAS cria registros (*logs*) da atuação de cada agente e da evolução dos preços dos leilões monitorados por este. Estes registros podem ser facilmente analisados em planilhas eletrônicas através da geração de gráficos como os apresentados nas figuras 6.1, 6.2 e 6.3. Estes gráficos bem como os respectivos registros (listados no Anexo B) comprovam ainda a facilitação do trabalho de avaliar estratégias implementadas com o uso do AAS.

Realizou-se uma pequena análise comparativa entre as estratégias implementadas e pode-se perceber que a estratégia RT se mostra mais adequada aos usuários com fortes restrições de

tempo, enquanto a estratégia Greedy se mostra adequada aos usuários preocupados em obter o item ao menor preço possível e que possam se limitar apenas a leilões do tipo inglês.

7.2. Trabalhos Futuros

Apresenta-se aqui uma série de sugestões para futuros trabalhos, que podem não somente expandir as funcionalidades e aplicações do AAS, mas também contribuir para o aumento da automatização, facilidade de uso e confiabilidade de sistemas de agentes arrematantes. As sugestões são listadas abaixo, seguidas de alguns comentários .

- **Leilões Multiatributos:** O sistema AAS pode ser expandido para tratar leilões com vários atributos, ao invés de apenas preço. Esta abordagem permite a realização de negociações colaborativas (conforme descrito na seção 1.2) e que sejam considerados, por exemplo, características do fornecedor, como: reputação, prazos de entrega, qualidade, etc. Isto pode ampliar significativamente as aplicações do sistema AAS.
- **Leilões reversos:** Conforme explanado na seção 2.4, ter os arrematantes como compradores ou vendedores não altera significativamente as estratégias utilizadas nos agentes arrematantes. Entretanto, contemplar leilões reversos pode aumentar significativamente a abrangência do sistema.
- **Mecanismos de segurança:** O eMediator, talvez por tratar-se de um sistema exclusivamente acadêmico, não utiliza algoritmos de criptografia para proteger sua comunicação com os agentes, nem mesmo para as senhas. Embora isto não seja preocupante para um sistema acadêmico seria

fundamental estudar a utilização de mecanismos de segurança, caso se pretenda o uso comercial, ou mesmo para alguns usos acadêmicos, como competição de agentes arrematantes de diferentes autores, por exemplo. Da mesma forma, seria interessante aprofundar o estudo sobre a segurança nas comunicações feitas utilizando o sistema SACI.

- **Mecanismos de Comunicação baseados em “Web Services”:** Muitas aplicações de comércio eletrônico estão adotando protocolos de comunicação baseados em XML, muitas vezes tratados pelo termo “Web Services”. Para garantir a integração do sistema AAS com outros sistemas (por exemplo, sítios de leilões na Internet), seria interessante estudar a viabilidade do uso de protocolos deste tipo, como base para a comunicação no AAS.
- **Inclusão de Novos Servidores de Leilões** (em especial servidores de código aberto, se disponíveis): O sistema AAS foi desenvolvido de modo a facilitar o trabalho com vários servidores de leilões, pois contemplar vários servidores (e conseqüentemente maior número de leilões) não apenas aumenta as opções de compra para o usuário, mas também aumenta o valor agregado por agentes atuantes em múltiplos leilões. Para isto, seria necessário desenvolver apenas novos processos Comunicadores (ServerCommunicator). Nota-se que devido à flexibilidade do sistema a estrutura de comunicação entre os outros agentes e os demais processos Comunicadores não seria alterada.
- **Aprimoramento do Mecanismo de Identificação de Produtos:** Seria interessante estudar ainda mecanismo de identificação não ambígua de produtos, seja através de descrições textuais, do uso de ontologia unificada ou

através do uso de múltiplas ontologias, pois virtualmente cada vendedor/fabricante pode ter uma ontologia própria para descrever seus produtos.

7.3. Considerações Finais

Este trabalho abordou aspectos importantes de três áreas de estudo distintas: teoria de leilões, agentes autônomos e sistemas de leilões baseados em agentes. Observou-se uma lacuna na pesquisa de agentes para a atuação em múltiplos leilões simultâneos, que era a falta de uma infra-estrutura para o desenvolvimento de agentes arrematantes. Objetivando sanar esta deficiência, foi projetado e implementado um sistema chamado AAS. Com o uso deste sistema, o desenvolvimento e avaliação de novas estratégias para atuação em múltiplos leilões simultâneos serão facilitados. Isto ocorre devido ao oferecimento de uma interface de mais alto nível para lidar com um servidor de leilões eletrônicos (eMediator) e a disponibilidade de serviços para a busca e monitoramento de leilões, o que liberta o projetista de agentes de tais tarefas, liberando-o para se dedicar ao desenvolvimento das estratégias de atuação. Desta forma, este trabalho pode contribuir para o avanço na pesquisa de novas estratégias de atuação de agentes arrematantes autônomos, especialmente para o caso de múltiplos leilões simultâneos.

ANEXO A - Comandos AAS

Tabela A.1. Sintaxe dos Comandos AAS (BNF)

<p>Comando ALL = pedido resposta</p> <p>pedido := request=request_command&pairs</p> <p>request_command := create_bid get_auction_info search_auction list_auction monitor_auction set_auction_info remove_auction_search remove_auction</p> <p>resposta := status_request(&message)(&pairs)</p> <p>status_request := status=Ok status=Error</p> <p>message := msg=value</p> <p>pairs := pair pair&pairs</p> <p>pair := parameter=value parameter=value,value</p> <p>values := value value,value</p> <p>parameter := (a-zA-Z)(a-zA-Z0-9)*</p> <p>value := (a-zA-Z0-9%.)*</p>

Tabela A.2. Caracteres Especiais e seqüências de Escape nos comandos AAS

Caractere de Controle	Seqüência de Escape
%	%25
\n	%0
&	%26
=	%3d
,	%2c

Tabela A.3 Lista de Comandos AAS

Comando	Descrição	Origem	Destino
search_auction	Solicitação de busca dos leilões que negociam um determinado produto.	BidderAgent	AuctionSearcher
list_auction	Solicitação de lista de leilões abertos.	AuctionSearcher	ServerCommunicator
monitor_auction	Solicita a monitoração de um leilão.	AuctionSearcher	AuctionMonitor
get_auction_info	Solicita informações sobre um leilão.	AuctionMonitor	ServerCommunicator
set_auction_info	Envia informações sobre um leilão.	AuctionMonitor	BidderAgent
create_bid	Submete um lance.	BidderAgent	ServerCommunicator
remove_auction_search	Informa que um produto, cuja busca foi previamente solicitada (search_auction), não é mais de interesse do agente e portanto não se deve procurar por novos.	BidderAgent	AuctionSearcher

	leilões deste produto.		
List_bid	Obtém uma lista dos lances de um determinado agente.	BidderAgent	ServerCommunicator
remove_auction	Informa que um determinado leilão(ou uma lista de leilões), não é mais de interesse do agente e portanto é desnecessário continuar a monitorá-lo.	BidderAgent	AuctionMonitor

Tabela A.4. Parâmetros dos Comandos AAS

Comando	Parâmetros		Resposta
	Nome	Tipo	
search_auction	product_name	Texto, 40 caracteres	Ok Error
list_auction	Sem parâmetros		Lista de Leilões, ver tabela A.3 Error
monitor_auction	auc_list	Texto, 100 caracteres	Ok Error
	agent_name	Texto, 40 caracteres	

get_auction_info	auc_iden	Texto, 10 caracteres	Informações sobre Leilão, ver tabela A.5 Error
set_auction_info	Todos os parâmetros de um Leilão, ver tabela A.5.		-
create_bid	Todos os parâmetros de um lance, ver tabela A.4 e identificação do usuário proprietário do agente(usr_iden,usr_pass).		Ok Error
remove_auction_search	product_name	Texto, 40 caracteres	Ok Error
list_bid	usr_iden	Texto, 10 caracteres	Lista de Lance, ver tabela A.3 Error
	usr_pass	Texto, 10 caracteres	
remove_auction_monitor	auc_list	Texto, 100 caracteres	Ok Error

Tabela A.5. Parâmetros de uma Lista de Leilões e de uma lista de lances . Os parâmetros desta tabela têm o mesmo significado utilizado na tabela A.5. Cada parâmetro apresenta o mesmo número de valores, tal número corresponde ao número de leilões na lista.

	Parâmetros	
Resposta ao comando	Nome	tipo
list_auction	Auc_iden	Texto, 10 caracteres
	Auc_name	Texto, 60 caracteres

	Auc_desp	Texto
	Auc_date	Data(formato YYYYMMDDHHMMSS)
Resposta ao	Nome	Tipo
comando	Bid_iden	Texto, 10 caracteres
list_bid	Bid_name	Texto, 60 caracteres
	Bid_date	Data (formato YYYYMMDDHHMMSS)

Tabela A.6. Parâmetros dos Lances no AAS

Parâmetro de Lances no AAS			
No.	Parâmetro	Descrição	Domínio
1	auc_iden	Identificador do leilão ao qual o lance é submetido	Texto, tamanho 10 caracteres
2	bid_date	Data de criação do lance	Data (formato YYYYMMDDHHMMSS)
3	bid_iden	Identificador do lance	Texto, tamanho 10 caracteres
4	bid_name	Nome do lance (opcional)	Texto, tamanho 60 caracteres
5	bid_stat	Estado atual do lance	Enumeração: act - Lance Ativo cls - Lance Fechado (Leilão

			fechado) can - Lance Cancelado.
6	bid_valu e	Valor do lance	Número Real
7	exp_date	Data de expiração do lance	Data (formato YYYYMMDDHHMMSS)
8	exp_trig	Regra de expiração do lance	Enumeração: ful - Lance expira ao ser superado por outro lance fix - Lance expira em determinada data.
9	usr_iden	Identificador do usuário responsável pelo lance	Texto, tamanho 10 caracteres

Tabela A.7. Parâmetros de um Leilão no AAS

Parâmetros de um Leilão no AAS			
No	Parâmetro	Descrição	Domínio
1	auc_date	Data e Hora de abertura do leilão	Data (formato YYYYMMDDHHMMSS)
2	auc_desp	Descrição textual do Leilão	Texto
3	auc_iden	Identificador único do leilão	Texto, tamanho 10 caracteres
4	auc_mini	Incremento mínimo ao melhor	Número Real

		lance atual, usado em leilões ingleses	
5	auc_name	Nome do leilão (opcional)	Texto, tamanho 60 caracteres
6	auc_side	Ver capítulo 2 da dissertação para maiores informações sobre Leilões Simples e Leilões Duplos	Enumeração: sgl - Leilão Simples dbl - Leilão Duplo.
7	auc_stat	Estado atual do Leilão	Enumeração: act - Leilão Ativo cls - Leilão Fechado sus - Leilão Suspenso.
8	auc_urlx	Endereço Internet para maiores informações sobre o leilão (opcional)	Texto, tamanho 60 caracteres
9	bid_canc	Regra de cancelamento de Lance antes de ser declarado o vencedor do leilão, se é ou não permitido	Enumeração alw - Cancelamento Permitido not - Cancelamento não permitido.
10	bid_dcmt	Regra de cancelamento de Lance após ser declarado o vencedor do leilão, se é ou não permitido	Enumeração alw - Cancelamento Permitido not - Cancelamento não permitido.
11	bid_last	Data do Último lance	Data (formato YYYYMMDDHHMMSS)
12	bid_num	Número de lances recebidos pelo	Número Inteiro

		leilão	
13	bid_tiex	Regra para definição de empates entre lances	Enumeração rdm - Escolhe aleatoriamente um dos lances crt - Primeiro lance criado prevalece upd - Primeiro lance atualizado prevalece.
14	clr_inte	Intervalo para atualização do leilão	Dia, hora e minuto (formato 000000DDHHMM00)
15	clr_last	Data da última atualização das informações sobre o leilão	Data (formato YYYYMMDDHHMMSS)
16	clr_outx	Tempo de espera para atualização do leilão	Dia, hora e minuto (formato 000000DDHHMM00)
17	clr_trig	Regra de disparo da atualização do estado do leilão	Enumeração bid - Atualiza leilão quando novo lance chega int - Atualiza em um intervalo definido (clr_inte) out - Atualiza quando nenhum novo lance é recebido em determinado período (clr_outx) cls - Atualiza quando o leilão é fechado.

18	cls_date	Data de fechamento do leilão	Data (formato YYYYMMDDHHMMSS)
19	cls_trig	Regra de fechamento do leilão	Enumeração clr - Fecha o leilão quando da primeira atualização fix - Fecha o leilão em uma determinada data e hora cls - Fecha apenas sob comando do proprietário do leilão.
20	itm_list	Identificador do produto, no AAS um leilão negocia apenas um produto	Texto, tamanho máximo 40 caracteres
21	itm_resv	Preço mínimo para o produto	Número Real
22	prc_schm	Mecanismo de definição de preço	Enumeração 1st - Primeiro Preço vky - Leilão de Vickrey (Segundo Preço) Mid - Preço a ser pago é o valor médio entre oferta de compra e oferta de venda. Usado em leilões duplos.
23	bids_iden	Lista de identificadores dos lances apresentados no leilão e disponíveis para o público	Texto, tamanho máximo 100 caracteres

24	bids_value	Lista dos respectivos valores dos lances apresentados no leilão e disponíveis para o público, seguindo a ordem dada em bids_iden	Texto, tamanho máximo 100 caracteres
25	quo_date	Data da última atualização das informações sobre os lances	Data (formato YYYYMMDDHHMMSS)
26	quo_levl	Nível de informação disponibilizada em uma cotação	Enumeração not - Nenhuma informação é publicada bst - O melhor lance é apresentado quando oferecido pub - Todos os lances são públicos no momento em que são oferecidos clw - O melhor lance é público no momento da atualização do leilão cla - Todos os lances são públicos no momento da atualização do leilão.
27	usr_iden	Identificador do proprietário do leilão	Texto, tamanho máximo 8 caracteres

ANEXO B - Experimentos Realizados

Neste anexo, serão apresentados os “logs” com os lances e a evolução dos preços correntes dos leilões monitorados para experimento. Em todos os experimentos apresentados, o formato dos “logs” e obedece ao seguinte padrão:

Linha de log: (INSTANTANEO) (INFO_SOBRE_AGENTE) (INFO_SOBRE_LEILAO)*
(DATA)

INSTANTANEO: =>SS (Número seqüencial do instantâneo(Snapshot, SS))

INFO_SOBRE_AGENTE: (Identificador do leilão alvo do agente)TAB(Valor do lance do agente)

INFO_SOBRE_LEILAO: (Identificador de Leilão) (Valor Corrente de Leilão)

DATA: Date:(Data e hora corrente)

Os campos são separados por caracteres de tabulação.

Experimento 1: Atuação do Agente Greedy

Neste experimento, três leilões são disputados simultaneamente por quatro agentes do tipo Greedy. Para simular a atuação de arrematantes que operam em um único leilão foram usados 5 agentes MonoAuctionAgent (2 no leilão 305, 2 no leilão 306 e 1 no leilão 307). Os quatro agentes Greedy atuavam simultaneamente nos três leilões citados. É apresentado na figura 6.1. o gráfico do agente “GreedyAgent3”, o qual foi levantado a partir do “log” gerado pelo AAS. Em seguida, apresentam-se os “logs” do quatro agentes Greedy que atuaram neste experimento. Os leilões foram configurados como do tipo inglês, com preço mínimo igual a 200 e incremento mínimo igual a 1.

Tabela B.1 – Log do Agente “GreedyAgent” – Experimento 1

=>SS 1	0								Date:
=>SS 2	305	201	305	201	306	201	307	201	Date:
=>SS 3	306	201	305	202	306	201	307	201	Date:
=>SS 4	306	201	305	202	306	202	307	201	Date:
=>SS 5	306	201	305	203	306	202	307	201	Date:
=>SS 6	307	201	305	203	306	203	307	201	Date:
=>SS 7	307	202	305	203	306	203	307	202	Date:
=>SS 8	307	202	305	204	306	203	307	202	Date:
=>SS 9	307	202	305	204	306	204	307	203	Date:
=>SS 10	307	202	305	205	306	204	307	203	Date:
=>SS 11	307	203	305	205	306	204	307	203	Date:
=>SS 12	307	203	305	206	306	204	307	203	Date:
=>SS 13	306	204	305	206	306	204	307	204	Date:
=>SS 14	307	204	305	206	306	205	307	204	Date:
=>SS 15	307	205	305	206	306	206	307	205	Date:
=>SS 16	305	206	305	206	306	206	307	206	Date:
=>SS 17	306	206	305	207	306	206	307	206	Date:
=>SS 18	306	206	305	207	306	207	307	206	Date:
=>SS 19	307	206	305	207	306	207	307	206	Date:
=>SS 20	305	207	305	207	306	207	307	207	Date:
=>SS 21	305	207	305	207	306	208	307	207	Date:
=>SS 22	307	207	305	208	306	208	307	207	Date:
=>SS 23	305	208	305	208	306	208	307	208	Date:

=>SS 24	305	208	305	208	306	209	307	208	Date:
=>SS 25	305	208	305	209	306	209	307	208	Date:
=>SS 26	305	208	305	209	306	210	307	209	Date:
=>SS 27	305	209	305	209	306	210	307	209	Date:
=>SS 28	307	209	305	210	306	210	307	209	Date:
=>SS 29	305	210	305	210	306	210	307	210	Date:
=>SS 30	305	210	305	211	306	210	307	210	Date:
=>SS 31	305	210	305	211	306	211	307	210	Date:
=>SS 32	305	210	305	211	306	211	307	211	Date:
=>SS 33	305	211	305	211	306	211	307	211	Date:
=>SS 34	306	211	305	212	306	211	307	211	Date:
=>SS 35	306	211	305	212	306	212	307	211	Date:
=>SS 36	306	211	305	212	306	212	307	212	Date:
=>SS 37	306	211	305	213	306	212	307	212	Date:

Tabela B.2 – Log do Agente “GreedyAgent1” – Experimento 1

=>SS 1	0								Date:
=>SS 2	305	204	305	204					Date:
=>SS 3	305	204	305	204	306	203	307	202	Date:
=>SS 4	305	204	305	204	306	204	307	203	Date:
=>SS 5	307	203	305	206	306	204	307	203	Date:
=>SS 6	307	204	305	206	306	205	307	204	Date:
=>SS 7	307	204	305	206	306	205	307	205	Date:

=>SS 8	305	206	305	206	306	206	307	206	Date:
=>SS 9	305	206	305	207	306	206	307	206	Date:
=>SS 10	305	206	305	207	306	207	307	206	Date:
=>SS 11	305	206	305	207	306	207	307	207	Date:
=>SS 12	305	206	305	207	306	208	307	207	Date:
=>SS 13	305	207	305	207	306	208	307	207	Date:
=>SS 14	305	207	305	208	306	208	307	207	Date:
=>SS 15	307	207	305	208	306	208	307	207	Date:
=>SS 16	307	207	305	208	306	208	307	208	Date:
=>SS 17	307	207	305	208	306	209	307	208	Date:
=>SS 18	305	208	305	208	306	209	307	208	Date:
=>SS 19	305	208	305	209	306	209	307	208	Date:
=>SS 20	307	208	305	209	306	209	307	208	Date:
=>SS 21	307	208	305	209	306	210	307	209	Date:
=>SS 22	305	209	305	209	306	210	307	209	Date:
=>SS 23	307	209	305	210	306	210	307	209	Date:
=>SS 24	305	210	305	210	306	210	307	210	Date:
=>SS 25	306	210	305	211	306	210	307	210	Date:
=>SS 26	307	210	305	211	306	211	307	210	Date:
=>SS 27	307	210	305	211	306	211	307	211	Date:
=>SS 28	307	210	305	212	306	211	307	211	Date:
=>SS 29	307	210	305	212	306	212	307	211	Date:
=>SS 30	307	211	305	212	306	212	307	211	Date:
=>SS 31	305	212	305	212	306	212	307	212	Date:
=>SS 32	305	212	305	213	306	212	307	212	Date:

Tabela B.3 – Log do Agente “GreedyAgent2” – Experimento 1

=>SS 1	0								Date:
=>SS 2	305	205	305	205	306	204			Date:
=>SS 3	305	205	305	205	306	204	307	203	Date:
=>SS 4	305	205	305	206	306	204	307	203	Date:
=>SS 5	305	205	305	206	306	205	307	204	Date:
=>SS 6	305	205	305	206	306	205	307	205	Date:
=>SS 7	305	205	305	206	306	206	307	206	Date:
=>SS 8	306	206	305	207	306	206	307	206	Date:
=>SS 9	306	206	305	207	306	207	307	206	Date:
=>SS 10	307	206	305	207	306	207	307	206	Date:
=>SS 11	307	206	305	207	306	207	307	207	Date:
=>SS 12	307	206	305	207	306	208	307	207	Date:
=>SS 13	307	206	305	208	306	208	307	207	Date:
=>SS 14	307	207	305	208	306	208	307	207	Date:
=>SS 15	307	207	305	208	306	208	307	208	Date:
=>SS 16	305	208	305	208	306	209	307	208	Date:
=>SS 17	307	208	305	209	306	209	307	208	Date:
=>SS 18	307	208	305	209	306	210	307	208	Date:
=>SS 19	307	208	305	209	306	210	307	209	Date:
=>SS 20	305	209	305	209	306	210	307	209	Date:
=>SS 21	305	209	305	210	306	210	307	209	Date:
=>SS 22	307	209	305	210	306	210	307	209	Date:
=>SS 23	305	210	305	210	306	210	307	210	Date:
=>SS 24	305	210	305	211	306	210	307	210	Date:

=>SS 25	306	210	305	211	306	210	307	210	Date:
=>SS 26	307	210	305	211	306	211	307	210	Date:
=>SS 27	305	211	305	211	306	211	307	211	Date:
=>SS 28	305	211	305	212	306	211	307	211	Date:
=>SS 29	305	211	305	212	306	212	307	211	Date:
=>SS 30	305	211	305	212	306	212	307	212	Date:
=>SS 31	305	212	305	212	306	212	307	212	Date:
=>SS 32	306	212	305	213	306	212	307	212	Date:

Tabela B.4 – Log do Agente “GreedyAgent3” – Experimento 1

=>SS 1	0								Date:
=>SS 2	305	205	305	205	306	204	307	203	Date:
=>SS 3	307	203	305	206	306	204	307	203	Date:
=>SS 4	307	204	305	206	306	205	307	204	Date:
=>SS 5	307	205	305	206	306	206	307	205	Date:
=>SS 6	305	206	305	206	306	206	307	206	Date:
=>SS 7	306	206	305	207	306	206	307	206	Date:
=>SS 8	306	206	305	207	306	207	307	206	Date:
=>SS 9	307	206	305	207	306	207	307	206	Date:

=>SS 10	305	207	305	207	306	207	307	207	Date:
=>SS 11	305	207	305	207	306	208	307	207	Date:
=>SS 12	305	207	305	208	306	208	307	207	Date:
=>SS 13	305	207	305	208	306	209	307	207	Date:
=>SS 14	305	207	305	208	306	209	307	208	Date:
=>SS 15	305	208	305	208	306	209	307	208	Date:
=>SS 16	307	208	305	209	306	209	307	208	Date:
=>SS 17	307	208	305	209	306	210	307	208	Date:
=>SS 18	305	209	305	209	306	210	307	209	Date:
=>SS 19	305	209	305	210	306	210	307	209	Date:
=>SS 20	305	209	305	210	306	210	307	210	Date:
=>SS 21	306	210	305	211	306	210	307	210	Date:
=>SS 22	306	210	305	211	306	211	307	210	Date:
=>SS 23	306	210	305	211	306	211	307	211	Date:
=>SS 24	306	210	305	212	306	211	307	211	Date:
=>SS 25	306	211	305	212	306	211	307	211	Date:
=>SS 26	307	211	305	212	306	212	307	211	Date:
=>SS 27	307	211	305	212	306	212	307	212	Date:
=>SS 28	307	211	305	213	306	212	307	212	Date:

Experimento 2: Atuação do Agente RT

Neste experimento, três leilões são disputados simultaneamente por dois agentes do tipo RT. Para simular a atuação de arrematantes que operam em um único leilão foram usados 6 agentes `MonoAuctionAgent` (2 em cada leilão). Os dois agentes RT atuavam simultaneamente nos três leilões criados. É exibido na figura 6.2 o gráfico unificado com a evolução dos preços correntes dos três e do agente “RTAgent”, o qual foi levantado a partir do “log” gerado pelo AAS. Em seguida, apresentam-se os “logs” dos agentes “RTAgent” e “RTAgent1” que atuaram neste experimento. Salienta-se que nos agentes do tipo RT foram utilizados constantes k e β iguais a 0,5 (zero vírgula cinco), na fórmula (1) usada para calcular o valor do lance, citada na seção 6.3. Os leilões foram configurados como do tipo inglês, com preço mínimo igual a 200 e incremento mínimo igual a 1. A probabilidade de vencer o leilão é calculada pelo agente RT como a razão entre o valor do lance e o preço reservado limitada a 90% para leilões dos tipos inglês, americano e leilão de Vickrey. Para leilão holandês, tal probabilidade é calculada como a razão entre o preço reservado e o valor corrente do leilão, com valor limitado a 90%.

Tabela B.5 – Log do Agente “RTAgent” – Experimento 2

=>SS 1		0		Date:					
=>SS 2	326	0	326	200	327	200	328	200	
=>SS 3	326	0	326	201	327	201	328	201	Date:
=>SS 4	326	0	326	202	327	202	328	201	Date:
=>SS 5	326	0	326	202	327	202	328	202	Date:
=>SS 6	326	203	326	203	327	203	328	202	Date:
=>SS 7	326	203	326	203	327	203	328	203	Date:

=>SS 8	326	203	326	204	327	204	328	203	Date:
=>SS 9	326	206	326	206	327	204	328	204	Date:
=>SS 10	326	206	326	206	327	205	328	204	Date:
=>SS 11	326	206	326	206	327	205	328	205	Date:

Tabela B.6 – Log do Agente “RTAgent1” – Experimento 2

=>SS 1	0								Date:
=>SS 2	326	0	326	200	327	200	328	200	
=>SS 3	326	0	326	201	327	201	328	201	Date:
=>SS 4	326	0	326	202	327	202	328	201	Date:
=>SS 5	326	0	326	202	327	202	328	202	Date:
=>SS 6	326	202	326	203	327	203	328	202	Date:
=>SS 7	326	202	326	203	327	203	328	203	Date:
=>SS 8	326	204	326	204	327	204	328	203	Date:
=>SS 9	326	204	326	206	327	204	328	204	Date:
=>SS 10	326	204	326	206	327	205	328	204	Date:
=>SS 11	326	204	326	206	327	205	328	205	Date:

Experimento 3: Comparação de Atuação dos Agentes Greedy e RT

Neste experimento, três leilões são disputados simultaneamente por dois agentes dos tipos Greedy e RT. Para simular a atuação de arrematantes que operam em um único leilão foram usados 3 agentes MonoAuctionAgent (1 em cada leilão). Os agentes Greedy e RT atuavam simultaneamente nos três leilões citados. É apresentado na figura 6.3. o gráfico unificado com a evolução dos preços correntes dos três e dos agentes Greedy e RT, o qual foi levantado a partir do “log” gerado pelo AAS. Em seguida, apresentam-se os “logs” dos agentes Greedy e RT que atuaram neste experimento. Salienta-se que nos agentes do tipo RT foram utilizados constantes k e β iguais a 0,5 (zero vírgula cinco), na fórmula (1) usada para calcular o valor do lance, citada na seção 6.3. Os leilões foram configurados como do tipo inglês, com preço mínimo igual a 200 e incremento mínimo igual a 1.

Tabela B.7 - Log do Agente “GreedyAgent” – Experimento 3

=>SS 1		0			Date:				
=>SS 2	330	201	330	201	331	201	332	201	Date:
=>SS 3	331	201	330	202	331	201	332	201	Date:
=>SS 4	331	201	330	203	331	202	332	201	Date:
=>SS 5	331	201	330	204	331	202	332	201	Date:
=>SS 6	332	201	330	204	331	202	332	201	Date:
=>SS 7	331	202	330	205	331	202	332	202	Date:
=>SS 8	332	202	330	206	331	203	332	202	Date:
=>SS 9	332	202	330	207	331	203	332	202	Date:
=>SS 10	332	202	330	207	331	203	332	203	Date:

=>SS 11	332	202	330	208	331	203	332	203	Date:
=>SS 12	331	203	330	208	331	203	332	203	Date:
=>SS 13	331	203	330	209	331	203	332	203	Date:
=>SS 14	331	203	330	210	331	204	332	203	Date:
=>SS 15	332	203	330	210	331	204	332	203	Date:
=>SS 16	331	204	330	210	331	204	332	204	Date:
=>SS 17	332	204	330	210	331	205	332	204	Date:
=>SS 18	332	204	330	210	331	205	332	205	Date:
=>SS 19	331	205	330	210	331	205	332	205	Date:

Tabela B.8 - Log do Agente “RTAgent” – Experimento 3

=>SS 1	0								Date:
=>SS 2	0	330	201						Date:
=>SS 3	0	330	201	331	201	332	201		Date:
=>SS 4	0	330	202	331	202	332	201		Date:
=>SS 5	0	330	202	331	202	332	202		Date:
=>SS 6	330	203	330	203	331	203	332	202	Date:
=>SS 7	330	203	330	203	331	203	332	203	Date:
=>SS 8	330	203	330	204	331	204	332	203	Date:
=>SS 9	330	206	330	206	331	204	332	204	Date:
=>SS 10	330	206	330	206	331	205	332	204	Date:
=>SS 11	330	206	330	206	331	205	332	205	Date:

LISTA DE REFERÊNCIAS

(Alvares; Sichman, 1997) Alvares, L.; Sichman, J. Introdução aos Sistemas Multiagentes. In: Jornada de Atualização em Informática (XVI JAI), 16, Brasília, 1997. **Anais**. Brasília: MEDEIROS, C. M.B. (Ed.), 1997. p.1-38.

(Anthony et al., 2001) Anthony, P.; Hall, W.; Dang, V. e Jennings, N.R. Autonomous agents for participating in multiple on-line auctions. In: IJCAI Workshop on E-business and the Intelligent Web, Seattle-WA, 2001. **Anais Eletrônicos**. Seattle-WA:IJCAI, 2001. Disponível em <<http://www.csd.abdn.ac.uk/ebiweb/programme.html>>. Acesso em: 6 de jun. de 2003.

(Booch; Rumbaugh; Jacobson, 1999) Booch, G., Rumbaugh, J., Jacobson, I. **The Unified Modeling Language User Guide**. New York:Addison Wesley, 1999. 482p.

(Bradshaw, 1997) Bradshaw, J. **An Introduction to Software Agents**. Cambridge: MIT Press, 1997. Cap. 1., p.3-46 : Software Agents.

(Byde; Preist; Jennings, 2002) Byde, A; Preist, C.; Jennings, N.R. Decision Procedures for Multiple Auctions. In: Autonomous Agent Multi-Agent Systems (AAMAS) , Bologna-Italy, 2002. **Proceedings**. Bologna: ACM Press, 2002. p.613-622, part 2.

(Byde, 2001) Byde, A. A dynamic programming model for algorithm design in simultaneous auctions. In: Workshop on E-Commerce (WELCOM '01), 2., Heidelberg-Germany. **Proceedings**. Heidelberg: Springer-Verlag, 2001. p. 152-163.

(Castro; Sichman, 2003) Castro, P.; Sichman, J. Uma Infra-Estrutura para Agentes Atuantes em Múltiplos Leilões Simultâneos. In: Encontro Nacional de Inteligência Artificial, 4., Campinas, 2003. **Anais do XXIII Congresso da Sociedade Brasileira de Computação**. Campinas, 2003. p. 2256-2265. 1 CD-ROM.

(Finin et al., 1994) Finin, T., Fritzon, R., McKay, D. e McEntire, R. KQML as an agent communication language. In: International Conference on Information and Knowledge Management (CIKM'94), 3., Baltimore-USA. **Proceedings**. New York: ACM Press, 1994. p.456-463.

(Franklin; Graesser, 1996) Franklin, S.; Graesser, A. Is It an Agent or Just a Program? A Taxonomy for Autonomous Agents. In: International Workshop on Agent Theories, Architectures and Languages, 3., Berlin, 1996. **Proceedings**. New York: Springer-Verlag, 1996. p. 21-35.

(Garcia; Lopes; Bentes, 2001) Garcia, A.; Lopes, A.; Bentes, C. Electronic Auction with autonomous intelligent agents: Finding opportunities by being there. **Revista Iberoamericana de Inteligência Artificial**. No. 13, p.45-52, 2001.

(Guttman; Maes; Moukas, 1998) Guttman,R.; Maes,P.; Moukas,A. Agent as Mediators in Electronic Commerce. **EM-Electronic Markets**. Vol. 8, N. 1, May 1998. Disponível em: <<http://www.electronicmarkets.org/modules/pub/view.php/electronicmarkets-203>>. Acesso em: 26 de jun. de 2003.

(Green; Whinston; Mas-Colell, 1995) Green, J.; Whinston, M.; Mas-Colell, A. **Microeconomic Theory**. New York: Oxford University Press. 1995. p. 219-242

(Huai; Sandholm, 2000) Huai, Q.; Sandholm, T. Nomad: Mobile Agent System for an Internet-Based Auction House. **IEEE Internet Computing** (Special issue on Agent Technology and the Internet). V. 4, N.2, p.80-86, Mar/Apr 2000.

(Hubner; Sichman, 2000) Hubner, J.; Sichman, J. SACI: Uma ferramenta para implementação e monitoração da comunicação entre agentes. In: Simpósio Brasileiro de Inteligência Artificial, Atibaia-SP-Brasil, 2000. **Open Discussion Track**. Atibaia: Monard, M.C. e Sichman, J.S. (Eds.), 2000. p. 47-56.

(Ito; Fukuta; Sycara, 2000) Ito, T.; Fukuta, T.; Sycara, K. BiddingBot: A multiagent support system for cooperative bidding in multiple auctions. In: Proceedings of the Fourth International Conference on Multi-Agent Systems(ICMAS'2000), 4., Boston-USA, 2000. **Proceedings**. Boston, 2000, p. 399-400.

(Klemperer, 1999) Klemperer, P. Auction Theory: A Guide to the Literature. **Journal of Economic Surveys**. V.13 , N.3, p.227-286, 1999.

(Maes; Guttman; Moukas, 1999) Maes, P.; Guttman, R.; Moukas, A. Agents that Buy and Sell. **Communications of the ACM**. V. 42, N.3, p.81- 91, 1999.

(Preist; Bartolini; Philips, 2001) Preist, C.; Bartolini, C.; Philips, I. Algorithm design for agents which participate in multiple simultaneous auctions. In: Agent Mediated Electronic Commerce, 3, Berlin, 2001. **Proceedings**. Berlin: Springer-Verlag, 2001. p. 139-154.

(Preist; Byde; Bartolini, 2001) Preist, C.; Byde A.; Bartolini, C. Economic dynamics of agents in multiple auctions. In: 5th International Conference on Autonomous Agents. Montreal, 2001. **Proceedings**. Montreal, 2001. p.545-551.

(Sandholm, 2000) Sandholm, T; eMediator: A Next Generation Electronic Commerce Server. In: International Conference on Autonomous Agents (AGENTS), 4., Barcelona-Spain, 2000. **Proceedings**. Barcelona, 2000. p.341- 348.

(Sandholm; Ferrandon, 2000) Sandholm, T.; Ferrandon, V. Safe Exchange Planner. In: International Conference on Multiagent Systems (ICMAS), Boston-USA, 2000. **Proceedings**. Boston, 2000. p. 255-262.

(Sandholm; Lesser, 1996) Sandholm, T.; Lesser, V. Advantages of a Leveled Commitment Contracting Protocol. In: National Conference on Artificial Intelligence (AAAI'96), 14., Portland-USA, 1996. **Proceedings**. Portland, 1996. p.126-133.

(Wurman; Wellman; Walsh, 1998) Wurman, P.; Wellman, M.; Walsh, W. The Michigan Internet AuctionBot: A configurable auction server for human and software agent. In: International Conference on Autonomous Agents (AGENTS), 2., Minneapolis-USA., 1998. **Proceedings**. Minneapolis: St. Paul, 1998. p. 301-308.