

Um Mecanismo de Integração entre os Simuladores de Mercado Financeiro AgEx e MASSES

Renan C. A. Alves¹, Paulo André L. de Castro², Jaime S. Sichman¹

¹Laboratório de Técnicas Inteligentes, Escola Politécnica, Universidade de São Paulo
Av. Prof. Luciano Gualberto, 158 travessa 3, 05508-970, São Paulo, SP, Brasil

²Divisão de Ciência da Computação, Instituto Tecnológico da Aeronáutica
Pç. Mal. Eduardo Gomes, 50, 12228-900, São José dos Campos, SP, Brasil
renanalves@usp.br, pauloac@ita.br, jaime.sichman@poli.usp.br

Abstract. *This paper presents an integration mechanism between two financial market simulation tools, independently developed: MASSES and AgEx. This mechanism allows agents designed in AgEx to operate among MASSES agents in the same market simulation. In this paper, we describe the design and implementation of the integration mechanism AgEx-MASSES, and present a set of experiments that shows how AgEx agents can be used as players in a simulation managed by MASSES. Results of simulated experiments in various settings dates are presented and discussed.*

Resumo. *Este artigo apresenta um mecanismo de integração entre duas ferramentas de simulação de mercado financeiro desenvolvidas de forma independentes: MASSES e AgEx. Este mecanismo permite que agentes desenvolvidos para operar no AgEx participem de simulações de mercado em conjunto com agentes MASSES. Neste trabalho, descrevemos o projeto e implementação do mecanismo de integração AgEx-MASSES, e apresentamos um conjunto de experimentos que mostra ser possível o uso de agentes AgEx como operadores em uma simulação gerenciada pelo MASSES. Resultados de experimentos simulados em várias configurações de datas são apresentados e discutidos.*

1. Introdução

A automação de administração de ativos financeiros pode ser um importante meio para testar algoritmos e técnicas adaptativas pesquisadas em Inteligência Artificial. Observa-se que os registros de comercialização de ativos financeiros são facilmente acessíveis na Internet. Além disso, mercados financeiros tem regras de atuação relativamente simples, porém elevada complexidade no processo de tomada de decisão em termos das informações que devem ser consideradas e nos seus graus de relevância. Tais características, além das possíveis aplicações práticas, motivaram diversos pesquisadores a desenvolver agentes que negociam ativos financeiros de forma autônoma em sistemas que simulam o mercado financeiro.

Dois exemplos de simuladores financeiros, desenvolvidos quase na mesma época porém de modo independente no Brasil, são os sistemas MASSES [AZEVEDO et

al. 2008], desenvolvido na PUC-RJ, e AgEx [CASTRO e SICHMAN 2009], desenvolvido conjuntamente pelo ITA e pela POLI/USP. Além do núcleo dos simuladores propriamente dito, no âmbito destas pesquisas foram desenvolvidos agentes capazes de negociar autonomamente em ambos os sistemas; entretanto, um agente desenvolvido em um destes sistemas não era capaz de operar no outro ambiente. Tal deficiência inspirou o desenvolvimento deste trabalho, com o objetivo de aumentar a sinergia na área no país.

Deste modo, o objetivo deste trabalho é desenvolver um mecanismo de integração entre estes dois simuladores MASSES e AgEx. Tal mecanismo permitirá que agentes desenvolvidos para o simulador AgEx possa competir com agentes desenvolvidos para o simulador MASSES como ambiente de simulação do mercado financeiro.

O simulador MASSES foi desenvolvido no âmbito de uma competição também chamada MASSES [AZEVEDO et al. 2008]. Ela compreende o confronto de vários agentes autônomos de administração de ativos, através do sistema de simulação homônimo, que visam o máximo retorno e se comunicam com os outros agentes de forma remota, através de Web Services [W3Schools 2010].

O simulador AgEx [CASTRO e SICHMAN 2009] é voltado a agentes autônomos de software e utiliza-se da plataforma JADE [BELLIFEMINE et al. 2007] para realizar a troca de informações entre agentes através de mensagens que utilizam padrões preconizados pela FIPA [FIPA 2011]. O AgEx permite também a integração com ambientes externos de execução de ordens, através da implementação de uma interface para estabelecer a comunicação entre os sistemas. Esta característica foi utilizada na construção deste trabalho.

O restante deste trabalho está organizado da seguinte forma. Na Seção 2, são apresentados os simuladores utilizados neste trabalho, além de brevemente descritos outros trabalhos relacionados. Em seguida, na Seção 3, descreve-se o mecanismo de integração proposto neste trabalho. Os resultados experimentais obtidos com a utilização do mecanismo e a análise de tais resultados é feita na Seção 4. Finalmente, os resultados do trabalho são analisados e discutidos, e algumas sugestões para trabalhos futuros são propostas na Seção 5.

2. Simuladores de Mercado Financeiro

Muitos trabalhos de pesquisa já foram realizados com foco na criação de mecanismos para automatizar a administração de ativos, utilizando diversas técnicas de Inteligência Artificial. Entre as técnicas de IA utilizadas, podem ser citadas redes neurais, aprendizado por reforço, sistemas multiagentes, arquiteturas BDI, raciocínio baseado em casos, e inteligência de enxame, entre outras [Castro 2009]. Independentemente da técnica utilizada, a avaliação de sistemas de administração automatizada de ativos requer a utilização de um software que simule o mercado financeiro, isto é, o ambiente no qual atuarão os agentes. Dentre os vários exemplos deste tipo de simulador, serão descritos TAC [TAC 2011] e PLAT [PLAT 2011], além dos dois simuladores MASSES e AgEx utilizados neste trabalho. Estes últimos foram selecionados como base deste trabalho por terem sido desenvolvidos no Brasil e por terem interfaces de programação públicas.

A seguir, respectivamente nas seções 2.1 e 2.2, são descritos os simuladores MASSES e AgEx, que apresentam grandes diferenças de modelagem entre si. O projeto do mecanismo de integração aqui proposto está intrinsecamente relacionado em conciliar as diferenças de modelagem entre tais simuladores. Na Seção 2.3, apresenta-se sucintamente outros exemplos de simuladores de agentes econômicos.

2.1. Simulador MASSES

No simulador MASSES, os agentes são modelados como entidades passivas e cada um deles é implementado como um servidor *WebService*, cujo endereço identifica o agente para o simulador. O *WebService* deve implementar uma classe abstrata da linguagem Java, fornecida pelos organizadores da competição, contendo dois métodos.

O primeiro método tem a função exclusiva de informar os parâmetros iniciais (datas de início e fim da simulação, montante financeiro disponível). O outro método é acionado no momento em que o simulador está pronto para receber ordens de compra ou venda referentes ao próximo período de simulação.

Acessando diretamente a base de dados do MASSES para extrair os dados julgados pertinentes, a estratégia do agente deve ser então executada neste segundo método, que retorna as ordens de compra e venda para os ativos disponíveis. É importante destacar que o simulador tem total controle sobre os instantes de execução de ordens.

Os agentes são cadastrados no simulador através de uma interface WEB, fornecendo o nome do agente e o URL do *WebService*, além de algumas opções. As competições também são administradas pela interface WEB, cujo cadastro é feito inserindo a data inicial, a data final e o montante com o qual cada agente participante inicia a competição, bem como a seleção dos agentes participantes, que são escolhidos através de uma lista.

2.2. Simulador AgEx

Neste simulador, os agentes também são classes da linguagem Java que devem implementar uma interface fornecida. Eles são definidos por um conjunto de parâmetros, tais como lista de ativos a serem negociados, montante iniciais deste ativos e a quantia em dinheiro sob administração do agente.

Além dos dados referentes aos agentes negociantes, é possível parametrizar a própria simulação através de um arquivo de configuração, destinado a descrever parâmetros do simulador em si. Podem ser definidos, por exemplo, taxas de corretagem, período da simulação, local de gravação dos dados de negociação, etc.

A simulação é controlada por um relógio interno e os agentes podem definir ordens de compra e venda, ou requerer informações sobre os ativos a qualquer momento, pois os agentes são entidades ativas. As ordens de compra e venda relativas a cada ativo são enviadas a um componente responsável pela realização e contabilização das ordens, chamado de *Broker*. O *Broker* tem o papel de interagir com o banco de dados. É possível configurar o simulador para operar com um *Broker Adaptador* que ao invés de utilizar um banco de dados, implemente a comunicação com algum mecanismo de cotações externo. Isto pode ser feito caso se deseje conectar o AgEx a uma bolsa real

(Figura 1). Neste trabalho, foi implementado um *Broker* adaptador para realizar a comunicação com o simulador MASSES, conforme detalhado na Seção 3.

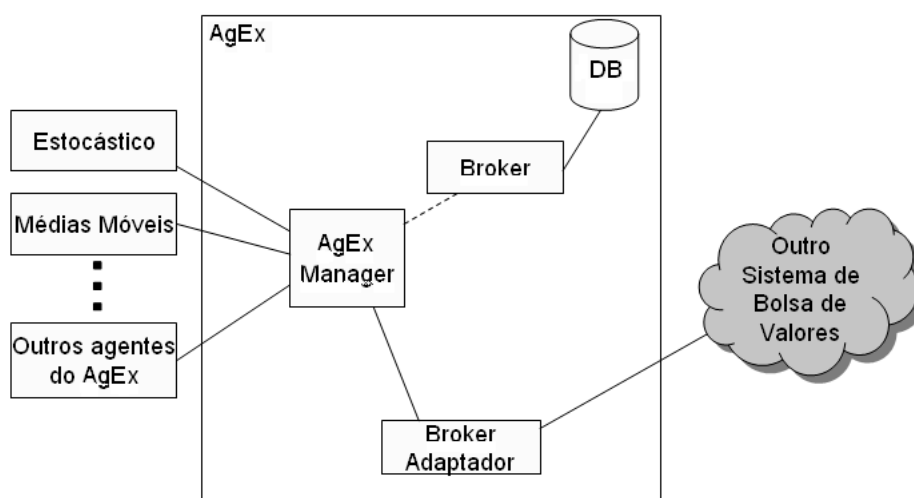


Figura 1. Arquitetura AgEx

2.3. Outros Simuladores

Podem ser encontrados na literatura vários simuladores e competições relacionadas a negociação automatizada de ativos por agentes. Entre estes outros trabalhos destacam-se a *Trading Agent Competition* (TAC) e o PLAT.

- **TAC:** A *Trading Agent Competition* (TAC) [TAC 2011] é uma competição com intuito de incentivar a pesquisa de agentes que comercializam bens¹. A competição tem duas vertentes, na primeira, denominada TAC Classic, os agentes competidores são agentes de viagem que buscam maximizar a satisfação dos clientes, e na outra, denominada TAC SCM, os agentes devem construir computadores baseados nos pedidos dos clientes, enquanto manejam as relações com os fornecedores de peças para montar estes computadores;
- **PLAT:** O *Penn-Lehman Automated Trading Project* (PLAT) [PLAT 2011] é um projeto que utiliza mercado financeiro como alvo de atuação dos agentes. O projeto utiliza um simulador de tempo real denominado *Penn Exchange Simulator* (PXS).

Como afirmado anteriormente, neste trabalho optamos por integrar os simuladores MASSES e AgEx por terem sido desenvolvidos no Brasil e por terem interfaces de programação públicas.

3. Mecanismo de Integração Proposto

Do ponto de vista do AgEx, é suficiente implementar uma classe *Broker* para concretizar a atuação dos agentes AgEx em algum outro sistema, conforme ilustrado na Figura 1.

¹ Neste caso, os ativos não são ações na bolsa, mas outro tipos de bens.

Todavia, o sistema com o qual se deseja interagir é o MASSES, que não é um prestador de serviços como seria esperado na construção de um Broker, mas sim outro sistema de simulação que não provê nenhum recurso, mas apenas utiliza os serviços dos agentes cadastrados.

Portanto, para estabelecer uma comunicação entre os sistemas, foi preciso desenvolver um intermediário entre o *Broker* do AgEx, denominado *MASSESBroker*, e o MASSES. Trata-se de um *WebService*, denominado *AgExMASSESAAdapter*, que proverá os serviços esperados para o AgEx e que também serve como um agente passivo para o MASSES (Figura 2).

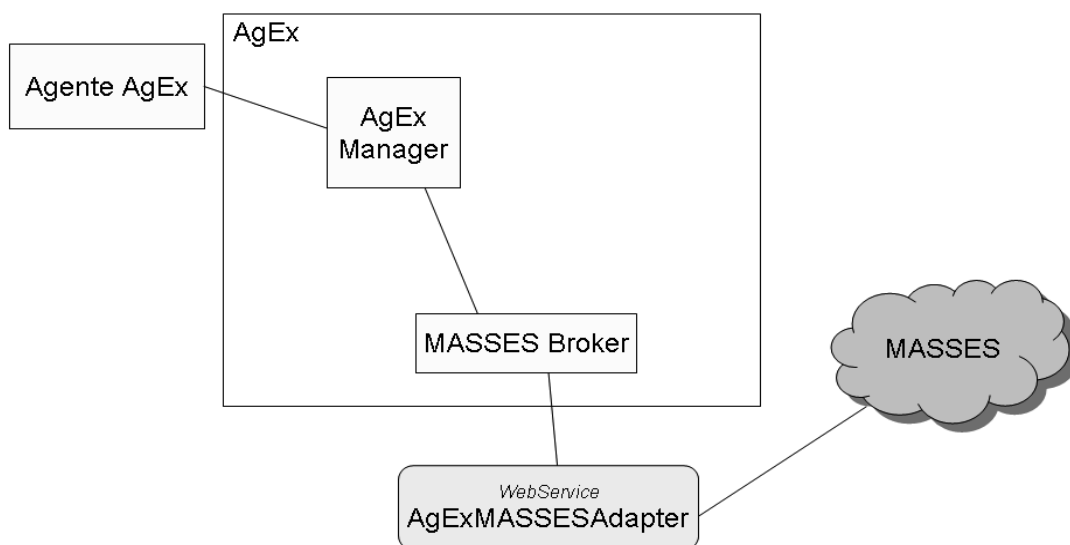


Figura 2. AgEx, Serviço AgExMASSESAAdapter e MASSES

Desta forma, todas as ordens de Compra/Venda de agentes provenientes do AgEx serão encaminhadas pelo *MASSESBroker* ao serviço *AgExMASSESAAdapter*, que as redireciona para o MASSES no momento em que for requerido. De forma semelhante, os pedidos para obter os dados de cotações de um ativo são enviados ao *AgExMASSESAAdapter*, que busca a informação desejada no banco de dados do MASSES.

Considerando a implementação, é importante ressaltar que o serviço *AgExMASSESAAdapter* deve utilizar o escopo *Application*, pois é necessário que o seu estado seja mantido para que as ordens fiquem armazenadas e sejam transmitidas de um simulador para o outro.

Do ponto de vista do MASSES, tudo se passa como se o simulador estivesse se comunicando com um agente comum, uma vez que foi implementada a classe abstrata de agente definida no site da competição. Por este motivo, cada serviço *AgExMASSESAAdapter* só pode traduzir as ordens de um AgEx com um único agente em execução, já que todo o AgEx é visto como um único agente pelo MASSES, as ordens de compra/venda seriam embaralhadas e não haveria distinção entre os agentes (Figura 3).

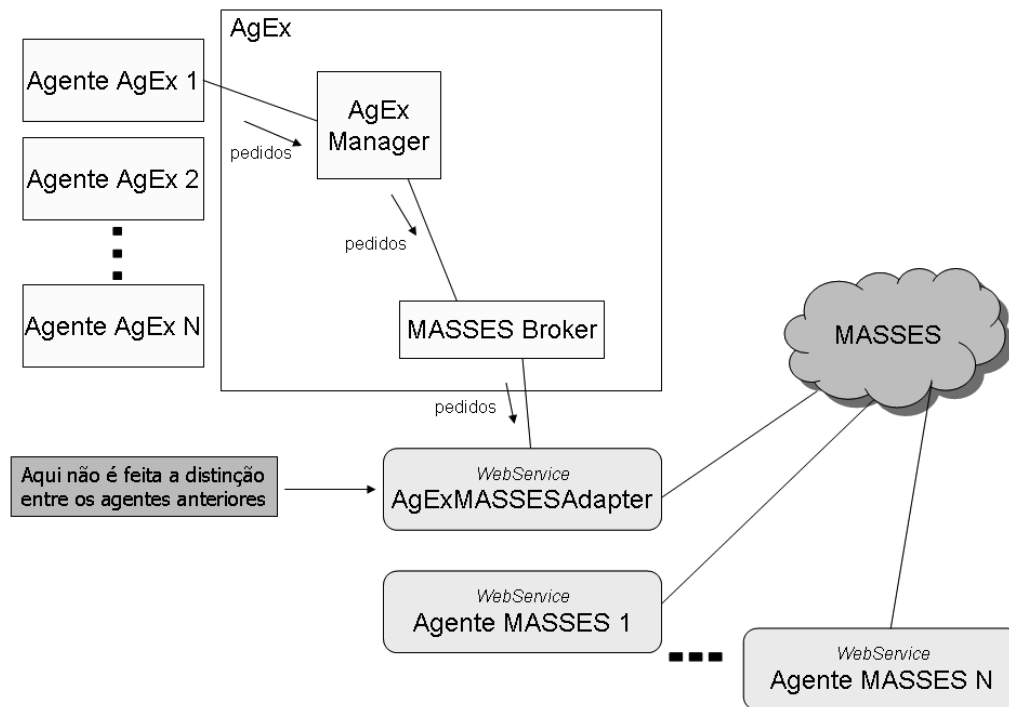


Figura 3. Conflito de Pedidos

Outro aspecto essencial na concepção do serviço AgExMASSESAadapter é a sincronia do tempo de simulação entre os sistemas, pois a princípio tanto o AgEx como o MASSES têm seu próprio relógio e contam o tempo de maneira independente.

Para contornar esta ambiguidade, foi adicionada ao AgEx a possibilidade de que o controle de tempo seja feito externamente. Este novo modo foi chamada de “*slave*”, o modo tradicional designado como “*master*”, e a escolha entre um ou outro modo é feita através do arquivo de configuração do simulador.

Além disso, outras modificações menores foram feitas para concretizar o controle do tempo de simulação pelo MASSES. O serviço AgExMASSESAadapter foi incumbido de armazenar o tempo mais recente enviado pelo MASSES e possui um método específico para que este valor possa ser recuperado pelo AgEx, mais especificamente, por uma thread específica da classe *MASSESBrokerFactory*, que provê instâncias de *MASSESBroker*. Esta classe esta foi escolhida porque ela é instanciada antes do início da simulação e segue o padrão Singleton, ou seja, há somente uma instância da classe para todo o sistema; assim, existirá somente uma entidade responsável por atualizar o tempo de simulação do simulador AgEx operando no modo “*slave*”.

O ciclo da interação entre os dois simuladores está resumido na Figura 4. O serviço AgExMASSESAadapter e o AgEx devem ser disparados antes do início da simulação. Após receber as informações iniciais, o serviço AgExMASSESAadapter está pronto para iniciar a simulação. A partir de cada indicação de novo período de simulação, é desencadeada uma sequência de eventos que resulta em ordens de

compra/venda enviadas ao simulador MASSES. Este ciclo se repete até que se esgotem os dias de simulação, conforme mostra a figure 4.

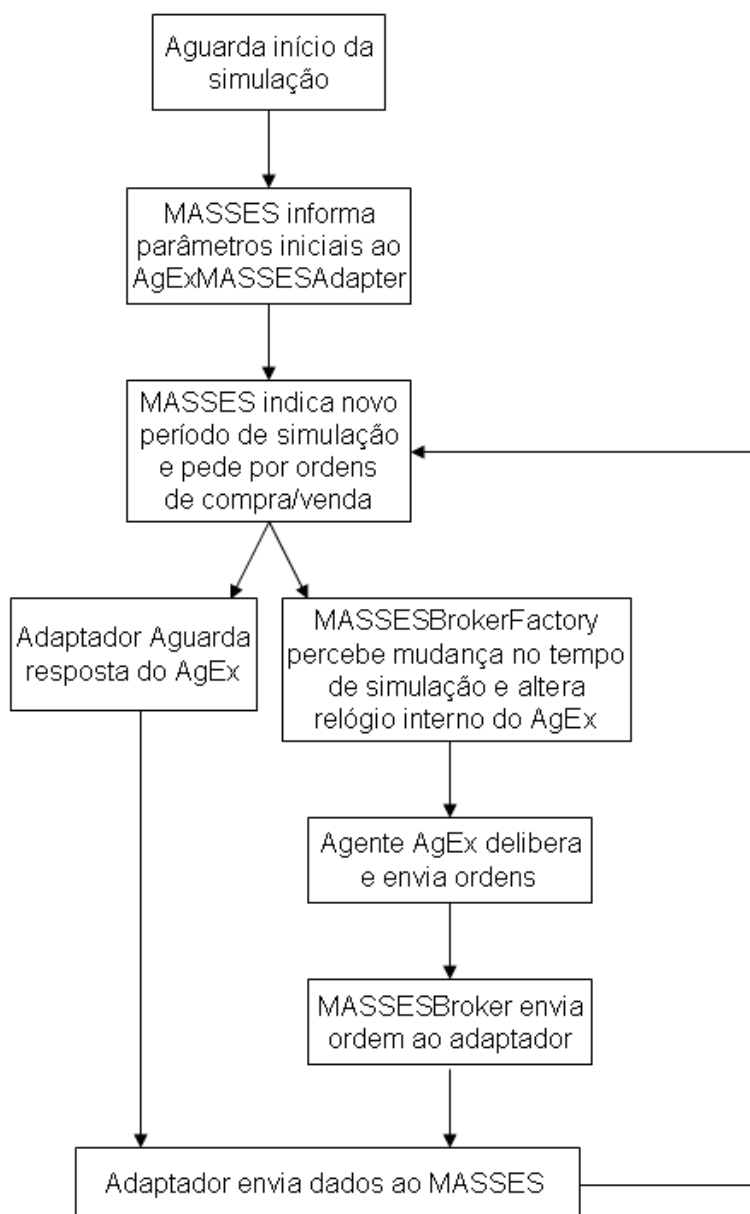


Figura 4. Ciclo de Simulação

4. Experimentos Realizados e Resultados Obtidos

Com o objetivo de validar o mecanismo de integração proposto, foi realizado um conjunto de experimentos simulados, colocando em um mesmo ambiente virtual agentes construídos para serem executados no AgEx e agentes construídos para operar no sistema MASSES.

Os agentes do primeiro tipo foram desenvolvidos na arquitetura COAST, implementada utilizando o simulador AgEx. Tal sistema consiste em um sociedade de agentes que administram ativos de forma automatizada compondo de forma negociada várias estratégias de análise de ativos [CASTRO 2009]. Ao contrário da maioria dos

trabalhos nesta área, comporta vários possíveis perfis de investidores, que visam quer a maximização de retorno, quer a minimização do risco, ou ainda uma relação de compromisso entre ambos. Uma descrição mais detalhada do sistema COAST está fora do escopo deste trabalho, maiores detalhes podem ser obtidos em [CASTRO 2009] [CASTRO e SICHMAN 2013].

A validação foi feita através de testes preliminares envolvendo agentes AgEx operando no ambiente MASSES, como descrito na Seção 4.1; e em uma segunda fase, no âmbito de uma competição entre a PUC-RJ e a POLI/USP, envolvendo agentes baseados no AgEx e também agentes desenvolvidos para operar de modo nativo no MASSES, como descrito na Seção 4.2.

4.1. Testes preliminares

Os testes descritos a seguir foram desenvolvidos para verificar se o comportamento de um agente AgEx atuando no MASSES, através do adaptador proposto por este trabalho, é funcionalmente semelhante ao de um agente nativo do MASSES. Para tal, foram escolhidos agentes de complexidade reduzida para testar aspectos específicos pertinentes à execução das competições. Estas foram implementados para ambos os simuladores, de forma que competições com as mesmas condições iniciais fossem realizadas.

O primeiro agente desta série, intitulado *BuyOnDates*, possui a estratégia de operação de comprar ou vender ativos em datas pré-determinadas. Esta estratégia não é realmente aplicada no mercado financeiro, porém é aplicável para verificar se as ordens são computadas no momento designado.

Na tabela 1, são apresentados os dias escolhidos para realizar as operações e o dia em que foram computadas nos dois casos estudados.

Tabela 1. Resultado do Agente *BuyOnDates*

Dia Escolhido Para Operação	Dia Em Que A Operação Foi Computada Pelo MASSES	
	Com Agente Nativo	Com Agente Agex
19/06/2006	19/06/2006	20/06/2006
20/06/2006	20/06/2006	21/06/2006
30/06/2006	30/6/2006	3/7/2006
02/07/2006	não foi computada	não foi computada
03/07/2006	03/07/2006	4/7/2006

Pelos resultados, observa-se que o agente nativo efetuou as operações nas datas pretendidas, exceto nos dias em que não há operação na Bolsa de Valores, o que ocorre no domingo referente à data de 02/07/2006. Portanto, o comportamento observado está de acordo com a lógica do simulador MASSES.

Entretanto, o agente AgEx teve suas ordens atrasadas em um dia útil, o que acarretou um atraso total de três dias no caso de sextas-feiras, como por exemplo a data

de 30/6/2006. Analogamente ao agente MASSES, as ordens que deveriam ser enviadas em dias em que não há operação na bolsa não são computadas.

Esta diferença de um dia é intrínseca a uma diferença de modelagem entre os dois simuladores em questão, pois no simulador AgEx especula-se sobre o preço das ações no próximo dia de simulação com base nas cotações do dia corrente e dos dias anteriores, enquanto que no MASSES especula-se sobre o valor dos papéis do dia corrente sobre os valores anteriores. Ambas as situações estão representadas na Figura 5, onde os quadros claros representam os dias em que há informação disponível, e os quadros escuros os dias em que não há tal informação.

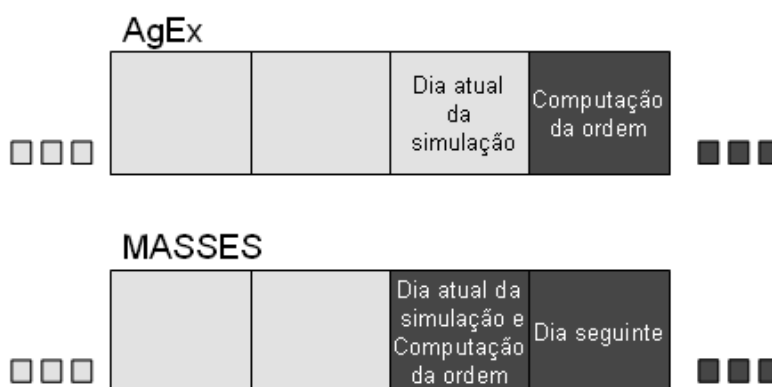


Figura 5. Diferenças de modelagem

Outro agente utilizado nestes testes preliminares foi o *GetCotations*, cuja estratégia é de realizar a recuperação do valor de cotações de papéis pré-definidos, ao invés de emitir ordens de compra e venda. Na tabela abaixo, estão os valores obtidos para os respectivos dias, onde se utilizou cotações da Petrobras (PETR4).

Tabela 2. Resultado do Agente GetCotations

Dia da Operação	Agente MASSES			Agente AgEx		
	Abertura	Máximo	Fechamento	Abertura	Máximo	Fechamento
21/06/2006	39.5	40.84	40.42	39.5	40.84	40.42
23/06/2006	40.21	41.55	41.05	40.21	41.55	41.05
07/07/2006	43.45	43.45	42.8	43.45	43.45	42.8
10/07/2006	42.82	42.95	42.62	42.82	42.95	42.62
11/07/2006	42.6	43.87	43.79	42.6	43.87	43.79

Como se pode observar, os valores obtidos são idênticos para os dois agentes. Esta fato implica que, para dada uma estratégia ou análise técnica determinística que utiliza como dados de entrada somente o valor das cotações, as ordens de compra/venda geradas por tal estratégias serão as mesmas tanto para agentes AgEx como para agentes MASSES.

Para confirmar esta conjectura, foi implementado um agente que utiliza o indicador estocástico [IE 2011] e foram observadas as ordens de compra e venda

emitidas pelos agentes MASSES e AgEx no mesmo período de simulação para os papéis da Usiminas (USIM5), Companhia Vale do Rio Doce (VALE3 e VALE5) e Telemar (TNLP4).

Como pode ser notado na tabela 3, as ordens são as mesmas, porém atrasadas de um dia, como discutido anteriormente para o caso do agente BuyOnDates. O agente MASSES possui, portanto, um número de ordens superior: estas ordens foram enviadas no último dia de simulação e não são realizadas pelo agente AgEx, pois seriam contabilizadas com um dia de atraso, numa data posterior à data final da simulação.

Tabela 3. Resultado do Agente Estocástico

Agente MASSES			Agente AgEx		
Dia	Ativo	Quantidade	Dia	Ativo	Quantidade
11/7/2006	VALE5	1000			
11/7/2006	TNLP4	1000			
11/7/2006	TNLP4	-3000	11/7/2006	TNLP4	1000
11/7/2006	USIM5	-5000	11/7/2006	VALE3	1000
11/7/2006	VALE3	-5000	11/7/2006	VALE5	1000
11/7/2006	VALE3	-1000	11/7/2006	TNLP4	-4000
10/7/2006	TNLP4	1000	11/7/2006	USIM5	-1000
10/7/2006	VALE3	1000	11/7/2006	VALE3	-3000
10/7/2006	VALE5	1000	11/7/2006	VALE5	-4000
7/7/2006	TNLP4	1000	10/7/2006	TNLP4	1000
7/7/2006	USIM5	1000	10/7/2006	USIM5	1000
7/7/2006	VALE3	1000	10/7/2006	VALE3	1000
7/7/2006	VALE5	1000	10/7/2006	VALE5	1000
6/7/2006	TNLP4	1000	7/7/2006	TNLP4	1000
6/7/2006	VALE3	1000	7/7/2006	VALE3	1000
6/7/2006	VALE5	1000	7/7/2006	VALE5	1000
5/7/2006	TNLP4	1000	6/7/2006	TNLP4	1000
5/7/2006	VALE3	1000	6/7/2006	VALE3	1000
29/6/2006	TNLP4	-3000	30/6/2006	TNLP4	-3000
28/6/2006	TNLP4	1000	29/6/2006	TNLP4	-1000
28/6/2006	USIM5	-1000	29/6/2006	USIM5	1000
28/6/2006	VALE3	-1000	29/6/2006	VALE3	-1000
27/6/2006	TNLP4	1000	28/6/2006	TNLP4	1000
27/6/2006	USIM5	1000	28/6/2006	USIM5	1000
27/6/2006	VALE3	1000	28/6/2006	VALE3	1000
26/6/2006	TNLP4	1000	27/6/2006	TNLP4	1000

Os diversos testes descritos nesta seção nos deram confiança no funcionamento do mecanismo de integração entre AgEx e MASSES para realizar uma competição de agentes desenvolvidos para estes dois simuladores utilizando o mecanismo de integração proposto neste trabalho. Tal competição é descrita a seguir.

4.2. Competição MASSES

A competição MASSES em sua edição 2010 foi realizada entre os dias 19 a 23 de julho de 2010 no Laboratório de Técnicas Inteligentes (LTI) da POLI/USP. Contemplou agentes desenvolvidos pela PUC-RJ, no ambiente MASSES, e por sociedades COAST, com cinco configurações diferentes, representando o LTI da POLI/USP.

As rodadas contemplaram 36 ativos financeiros negociados na BOVESPA, selecionados pela equipe da PUC-RJ, e períodos de tempo entre fevereiro de 2006 e outubro de 2008, com intervalos de 70 e 140 semanas. O resultado final da competição encontra-se na tabela 4.

Tabela 4. Competição MASSES - Classificação Final

Agente	Pontuação
Madoff	52
Tendências	49
Buffet	46
Soros	40
AgenteMasses	38
Poseidon	36
COAST_Config5	35
COAST_Config2	32
COAST_Config4	24
COAST_Config1	21
COAST_Config3	19
Barack	16

Os agentes COAST haviam sido desenvolvidos utilizando dados de negociação oriundos da Bolsa americana Nasdaq e simulador AgEx, enquanto os demais foram desenvolvidos utilizando dados da BOVESPA e o simulador MASSES. A competição foi realizada com dados da BOVESPA devido a maior facilidade de alterar os ativos alvos por parte daqueles agentes.

Apesar dos testes da seção anterior, durante a execução dos jogos notou-se que algumas ordens emitidas pelo COAST, através do AgEx, foram perdidas. Este problema ocorreu devido a existência de erros intermitentes no sincronismo entre os simuladores e o serviço AgExMASSESAdapter quando há uma quantidade grande de ativos e ordens para serem gerenciados. Apesar dos esforços despendidos para a resolução deste problema, não se chegou ainda a uma solução satisfatória. Apesar deste problema ter prejudicado o desempenho das sociedades COAST, não se espera que seus efeitos sejam drásticos, a ponto de alterar o resultado da competição.

5. Conclusões

Apesar dos problemas de sincronismo existentes, o serviço AgExMASSESAdapter cumpre bem sua função, uma vez que foi possível realizar competições utilizando sociedades COAST no simulador MASSES. Assim, este trabalho também serviu como prova de conceito deste serviço, já que permitiu ao simulador AgEx se comunicar com outro sistema.

Foi discutida ainda entre as equipes da PUC-RJ e POLI/USP a possibilidade de construção em parceria de um simulador de mercado financeiro mais completo e de fácil utilização, de modo a fomentar pesquisas e competições acadêmicas em administração automatizada de ativos. Observou-se que a integração de um simulador como este com iniciativas abertas a competidores humanos, tais como UOL Invest, poderia ser útil para popularizar o simulador e as pesquisas na área. O desenvolvimento de um simulador de mercado financeiro poderia ainda incentivar trabalhos de iniciação científica em Engenharia de Software e Inteligência Artificial.

Agradecimentos

Este trabalho foi fruto do projeto de Iniciação Científica do aluno Renan C. A. Alves, realizado no período de agosto/2009 a julho de 2010, na Universidade de São Paulo, com financiamento do CNPq (bolsa PIBIC). Jaime Simão Sichman é parcialmente financiado pelo CNPq e FAPESP, Brasil.

Referências

- AZEVEDO, Sérgio; ABREU NETTO, Manoel; COSTA, Andrew Diniz; BORSATO, B.; SORAES, F.; LUCENA, Carlos. Multi-Agent System for Stock Exchange Simulation – MASSES. In: *Proc. IV Workshop on Software Engineering for Agent-oriented Systems (SEAS 2008)*, SBES 2008, 13 - 17 de Outubro, Campinas, SP, Brasil, 2008, pp. 61-72.
- BELLIFEMINE, F. L.; CAIRE, G.; GREENWOOD, D. *Developing Multi-Agent Systems with JADE*. John Wiley & Sons, 2007.
- CASTRO, Paulo André Lima de; SICHMAN, Jaime Simão. Asset Management Based on Partially Cooperative Agents for a World of Risks. *Applied Intelligence*, 38(2), p. 210-225, 2013.
- CASTRO, Paulo André Lima de; SICHMAN, Jaime Simão. AgEx: A Financial Market Simulation Tool for Software Agents. In: *Joaquim Felipe José Cordeiro. (Org.). Enterprise Information Systems*. 1 ed. Berlin: Springer-Verlag, 2009, v. 24, p. 704-715.
- CASTRO, Paulo André Lima de. *Uma Arquitetura para Administração Automatizada de Ativos Baseada em Agentes Competitivos*. Tese de Doutorado. Escola Politécnica da Universidade de São Paulo. São Paulo. 2009
- FIPA 2011. The Foundation of Intelligent Physical Agents (FIPA). URL: www.fipa.org Acesso em Março. 2011.
- IE - Indicador Estocástico, <http://www.conexaodineiro.com/analise-tecnica/estocastico>. Acesso em Março. 2011.
- PLAT - THE PENN-LEHMAN AUTOMATED TRADING PROJECT. <http://www.cis.upenn.edu/~mkearns/projects/plat.html>. Acesso em Março. 2011.
- TAC - Trading Agent Competition - <http://www.sics.se/tac/>. Acesso em Março. 2011.
- W3Schools - W3Schools Web Services Tutorial - <http://www.w3schools.com/webservices/>. Acesso em Março. 2010.