

Uma Experiência com Agentes Inteligentes e Jogos de Cartas

Anita Maria da R. Fernandes
Universidade do Vale do Itajaí
anita.fernandes@univali.br

Daniel de Oliveira
Universidade do Vale do Itajaí
daniel@procibe.com.br

Helton Machado Kraus
Universidade do Vale do Itajaí
heltonkraus@univali.br

Pedro Edmundo Floriani
Universidade do Vale do Itajaí
pedro_edmundo@hotmail.com

Vital Pereira Dos Santos Jr.
Universidade do Vale do Itajaí
vital_junior@yahoo.com.br

RESUMO

Este artigo apresenta uma experiência no uso de agentes inteligentes, tendo como motivador o jogo de cartas. O protótipo foi elaborado por um grupo de alunos da disciplina de Inteligência Artificial (IA), aplicado em um curso de Mestrado em Computação, e resume-se no desenvolvimento de agentes que simulam quatro jogadores de cartas e um juiz das partidas. Os agentes seguem regras elaboradas a partir de uma árvore semântica, avaliam as jogadas e decidem de acordo com a concepção do sistema.

Palavras-Chave: Inteligência Artificial. Agentes Inteligentes. Sistema Especialista

1. INTRODUÇÃO

A tecnologia de agentes é um campo abrangente e tem adquirido, nos últimos anos, uma importância cada vez maior em muitos aspectos da computação, principalmente na área de Inteligência Artificial Distribuída. O conceito de autonomia e de aplicações capazes de executar tarefas de forma inteligente e independente tem despertando grandes interesses. Interações entre agentes que trabalham juntos para um objetivo único e maior, vêm reforçando e amadurecendo o conceito de sistemas multiagentes. (SILVA, 2003)

Os sistemas multiagentes, dentro do campo da Inteligência Artificial Distribuída não são apenas tópicos de pesquisa, mas já estão se tornando um importante assunto na área de ensino acadêmico e em aplicações comerciais e industriais. Como consequência disso, várias metodologias, arquiteturas e ferramentas já foram criadas para facilitar o desenvolvimento de sistemas multiagentes.

Para auxiliar no ensino-aprendizagem propõe-se um sistema multiagente naturalmente distribuído e cooperativo, com aplicação de um jogo. O uso de jogos no âmbito educacional é desejável, visto que pode auxiliar consideravelmente o professor, principalmente em relação à qualidade e ao tempo de aprendizagem do aluno, pois ajuda a tornar o ambiente favorável, como aponta GUILLON E MIRSHAWKA (1994): “Para se ter um aprendizado acelerado, é necessário que os fatores ambientais sejam observados e levados a sua melhor condição”.

O objetivo desta experiência, em que os jogadores não são humanos, é o aprendizado dos conceitos de Agentes Inteligentes e de Sistemas Especialistas, buscando desenvolver um protótipo que contemple estas duas técnicas de forma integrada, levantando formas de aplicação destas técnicas, onde foi apontado o jogo de truco como estudo de caso. Para o desenvolvimento foram utilizadas ferramentas apropriadas que atendessem a essa necessidade, possibilitando a criação de um protótipo que validasse esses conceitos.

Na experiência, os agentes foram desenvolvidos como sendo os atuadores, capazes de interagir entre si dando andamento ao jogo, tornando o sistema capaz de tomar decisões em função de regras, e permitindo a simulação de jogadores de cartas. As regras servem para organizar as ações, limitando o que pode ou não ser feito. O sistema especialista avalia a situação do jogo e define a decisão tática de cada agente na sua jogada.

Neste sentido, o artigo pretende mostrar quão interessante e produtiva podem ser as aulas, usando como base a elaboração de jogos pelos próprios alunos, e conseqüentemente o aprofundamento nas abordagens e técnicas desejadas. A Seção 2 apresenta a definição de agentes inteligentes, utilizado para elaboração do estudo de caso. O Sistema Especialista, que foi uma das técnicas de Inteligência Artificial aplicada, é discutido na Seção 3. A modelagem das regras do jogo é apresentada na Seção 4, seguida pela Seção 5 que esclarece sobre a arquitetura do sistema.

2. AGENTES INTELIGENTES

A definição de agente depende do ponto de vista do autor, e também da funcionalidade desse agente. Um agente pode ser um programa de computador, entretanto não precisa necessariamente apresentar comportamento “inteligente”, termo que é alvo de muita controvérsia, já que é difícil definir o que é realmente um comportamento inteligente (FERNANDES, 2003).

BOCCA, JAKES E VICARI (2003) entendem que agentes são resolvidores de problemas, autônomos e podem ser de natureza heterogênea. O fato é que não existe consenso de seu significado ou classificação, podendo ser desde simples processos de hardware e/ou software até realizadores de tarefas complexas. Porém, alguns aspectos da inteligência humana precisam ser representados no agente, para que ele seja reconhecido como tal.

De acordo com BARRETO (2001), um agente é um sistema dinâmico com capacidade de receber informações e agir sobre um ambiente, objetivando realizar uma determinada tarefa. Esta definição utiliza o modelo do Quadro-Negro (QN), com uma metáfora, onde cada agente é considerado um funcionário de uma fábrica de móveis. Assim como os funcionários, os agentes detêm conhecimentos específicos, compatíveis com sua função. O tesoureiro sabe gastar, receber e calcular, o chefe de fabricação sabe produzir os móveis, e assim por diante, ocorrendo dessa forma economia de recursos. Quando um funcionário tem necessidade de algo ele vai até o QN escrever o que deseja, o encarregado de compras lê o QN, faz a encomenda e escreve no QN o valor que precisa ser pago. O tesoureiro lê a mensagem, faz o pagamento e escreve no QN que a encomenda já está paga. O chefe do almoxarifado, após ver a mensagem, recebe a encomenda e escreve no QN. Por fim o funcionário que precisava da encomenda vai buscá-la, e a fábrica de móveis continua a funcionar.

Para RUSSEL E NORVIG (2004), um agente é aquele que percebe o seu ambiente por meio de sensores, e age sobre ele através dos atuadores. Num agente robótico os sensores poderiam ser câmeras de filmagem e detectores de faixa de infravermelho. Já os atuadores podem ser representados pelos motores e braços mecânicos.

A classificação de um agente pode ser feita tendo como base suas características ou propriedades básicas, sendo que quanto mais propriedades o agente possuir, maiores serão suas capacidades. As principais propriedades atribuídas aos agentes são: Autonomia, Mobilidade, Cooperação, Comunicabilidade, Aprendizagem, Reatividade, Habilidade Social e Pró-Atividade (FERNANDES, 2003).

Dessa forma, uma das características fundamentais em sistemas multiagentes é a comunicação, a qual permite a colaboração, negociação e a cooperação entre entidades

independentes, a fim de realizar um determinado conjunto de tarefas ou objetivos. Esses objetivos podem ser comuns a todos os agentes ou não.

Nos sistemas multiagentes, é necessário que a comunicação seja disciplinada para que os objetivos sejam alcançados de forma efetiva e eficientemente. Para isso, pode ser necessário que um dos agentes seja responsável pela coordenação do grupo, resultando assim, em um sistema coerente.

Agentes podem trocar mensagens diretamente, podem comunicar-se através de um agente “facilitador” especial através de um sistema “federado” (comunicação assistida), podem também utilizar uma comunicação por difusão de mensagens (*broadcast*) e até utilizar o modelo de quadro-negro (SILVA, 2003).

Os agentes, dentro de um sistema multiagente, podem ser heterogêneos ou homogêneos, colaborativos ou competitivos, etc. Ou seja, a definição dos tipos de agentes depende da finalidade da aplicação que o sistema está inserido (SILVA, 2003).

Neste artigo foi proposta uma comunidade de agentes, trabalhando sobre uma arquitetura multiagente, conforme definido por SILVA (2003). Para o protótipo desenvolvido, definiu-se que a comunicação seria realizada através de um agente facilitador, responsável pela coordenação do jogo.

3. SISTEMAS ESPECIALISTAS

Neste trabalho, os conceitos de sistemas especialistas foram utilizados para modelar as decisões táticas que um agente jogador pode tomar, em função dos eventos submetidos por outros jogadores.

3.1. DEFINIÇÃO DE SISTEMAS ESPECIALISTAS

Sistemas Especialistas, um ramo da Inteligência Artificial, são sistemas que emulam a capacidade de tomadas de decisão de um especialista humano. Tais sistemas atuam em domínios bem definidos de problemas e são constituídos por uma Base de Conhecimento, uma Memória de Trabalho e uma Máquina de Inferência.

A “inteligência” do especialista está contida na Base de Conhecimento, codificada em regras de produção; a Memória de Trabalho contém os fatos encontrados pelo sistema durante sua execução, e a Máquina de Inferência determina as conclusões ou ações a serem tomadas, cruzando os fatos da memória com a base de conhecimento.

3.2 SHELLS PARA SISTEMAS ESPECIALISTAS

Para o desenvolvimento de um Sistema Especialista necessita-se de um Motor de Inferência, usualmente fazendo o uso de *shells*. Os *shells* são softwares que simplificam o trabalho de desenvolvimento de sistemas especialistas, pois combinam fatos e regras de modos diversos, embutem uma máquina de inferência específica, assim como uma representação própria de conhecimento (MOURA E CRUZ, 2001).

Desta forma foram analisadas *shells* para desenvolvimento de sistemas especialistas, sendo apresentadas suas principais características:

- Expert Sinta: Esta ferramenta utiliza um modelo de representação do conhecimento baseado em regras de produção e probabilidades, permitindo a construção automática de telas e menus (LIA/UFC, 1996).
- CLIPS: uma ferramenta que fornece um ambiente completo para a construção de sistemas especialistas baseados em regras (RILEY, 2008).

- JClips: uma biblioteca de software, que permite aos programadores Java incorporarem aos seus programas o motor de inferência do CLIPS, através de mensagens em formato texto (MENKEN, 2008).
- JESS: É semelhante ao JCLIPS, porém não utiliza mensagens em texto, mas provê uma biblioteca que permite manipular os próprios objetos Java. No entanto, o JESS não é um software livre como os demais citados anteriormente (JESS, 2008).

No desenvolvimento deste trabalho as regras foram modeladas tanto na Shell Expert Sinta como em CLIPS. O Expert Sinta foi utilizado para testar e validar as regras dos agentes, antes delas serem inseridas no protótipo. O CLIPS foi utilizado para gerar o motor de inferência do agente propriamente dito.

Entre as bibliotecas analisadas, a escolhida para desenvolver o protótipo foi a biblioteca JClips, devido sua licença de uso livre e sua compatibilidade com a linguagem de programação Java, linguagem de maior domínio dos integrantes da equipe.

4. EXPERIÊNCIA REALIZADA

O objetivo desta experiência foi o aprendizado dos conceitos de Agentes Inteligentes e de Sistemas Especialistas, buscando desenvolver um protótipo que aplicasse estas técnicas de maneira integrada, alcançando assim o objetivo da disciplina.

Para a aplicação de sistemas especialistas e agentes foram levantadas formas de aplicação destas técnicas, onde foi apontado o jogo de truco como estudo de caso. No protótipo, os agentes foram desenvolvidos como sendo os atuadores do jogo, capazes de interagirem entre si dando andamento ao jogo. O sistema especialista, conforme apresentado na seção anterior, avalia a situação do jogo e define a decisão tática de cada agente na sua jogada.

4.1. MODELAGEM DAS REGRAS DO JOGO

Truco é um jogo de cartas onde o objetivo é você e seu parceiro, conquistarem 12 pontos. Cada ponto é conquistado através do método “melhor de três”, onde a equipe deve vencer ao menos duas rodadas (COPAG, 1990). O jogo inicia com a distribuição de três cartas para cada jogador e a definição de uma carta como manilha, normalmente mantida virada sobre a mesa, que é uma carta acima da carta virada. Este jogo normalmente é formado de duas equipes rivais com dois jogadores cada.

Cada rodada é concluída quando todos os jogadores jogam uma de suas cartas. A qualquer momento um jogador pode pedir truco, quando isso acontece os jogadores da equipe adversária visualizam as cartas do respectivo parceiro e avaliam se aceitam ou não o pedido. Caso seja aceito o pedido de truco, a partida passa a valer 3 pontos e a equipe que aceitou tem a oportunidade de “retruca”, onde a partida passa a valer 6 pontos. Caso um pedido de truco ou retruco não seja aceito, a equipe que recusou o pedido perde, cedendo os pontos atuais da partida para a outra equipe.

Como no jogo de truco as cartas têm um valor diferente da seqüência lógica de numeração, e ainda há um critério de desempate entre as manilhas, então se modelou os valores das cartas como mostra a Tabela 1. A tabela representa a seqüência das cartas, onde, na coluna “valor” aparece o valor propriamente dito, que foi utilizado como significância de cada carta nos cálculos do jogo de truco.

Tabela 1. Representação dos valores das cartas no jogo

Carta	Valor
4	1
5	2
6	3
7	4
10 (Q)	5
11 (J)	6
12 (K)	7
1	8
2	9
3	10
Manilha Ouro	11
Manilha Espadas	12
Manilha Copas	13
Manilha Paus	14

Desta forma, foram levantadas as necessidades do protótipo analisando-se as trocas de mensagens para o funcionamento das regras, bem como as interações dos agentes para o andamento do jogo, conforme apresentadas na seção seguinte.

5. ARQUITETURA DO SISTEMA

Como ferramenta para desenvolvimento de agentes inteligentes foi definido o JADE (JADE, 2008), que é um ambiente para desenvolvimento de aplicações baseado em agentes conforme as especificações da FIPA (*Foundation for Intelligent Physical Agents*) para interoperabilidade entre sistemas multiagentes totalmente implementado em Java (SILVA, 2003).

Como ponto de partida para o desenvolvimento do protótipo foram definidos os agentes que seriam necessários para as interações realizadas no jogo, chegando-se a conclusão de que dois tipos de agente seriam necessários:

- Agente Juiz: Responsável por sincronizar e intermediar a comunicação entre os agentes jogadores, solicitando a cada agente a execução de uma ação para o andamento do jogo;
- Agente Jogador: Responsável por interagir com Juiz apresentando as cartas escolhidas através do sistema especialista e também pedir truco.

Conforme apresentado, foram necessários um total de 5 agentes para a interação completa do jogo (1 juiz e 4 jogadores), sendo desta forma aplicado o conceito de Sistemas Multiagentes.

Para o protótipo desenvolvido foi definido que a comunicação seria realizada através de um agente facilitador (Juiz), responsável pela coordenação do jogo. Após a definição dos agentes foi realizada a modelagem das regras do jogo, permitindo que cada jogador seguisse uma estratégia, utilizando para modelagem redes semânticas. Na Figura 1 é apresentada a rede semântica utilizada para verificar se o jogador pedirá truco. Nesta árvore é possível constatar a regra “blefar”, onde se utiliza uma regra randômica com a possibilidade baixa de ser executada, para que o agente possa pedir truco, sem a prévia avaliação de suas cartas. Existem duas terminações: o pedido de truco ou a verificação das regras para jogar uma carta.

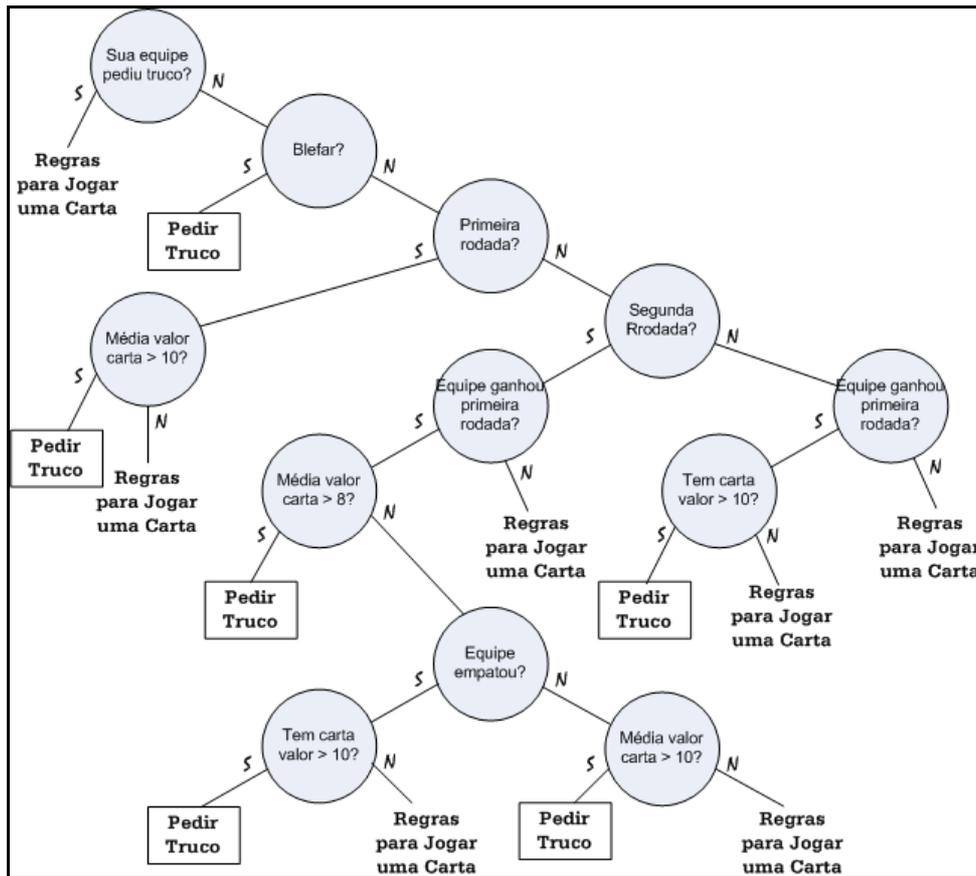


Figura 1. Rede semântica (Pedir Truco)

Na Figura 2 é apresentada a árvore semântica utilizada para seleccionar uma carta para ser jogada.

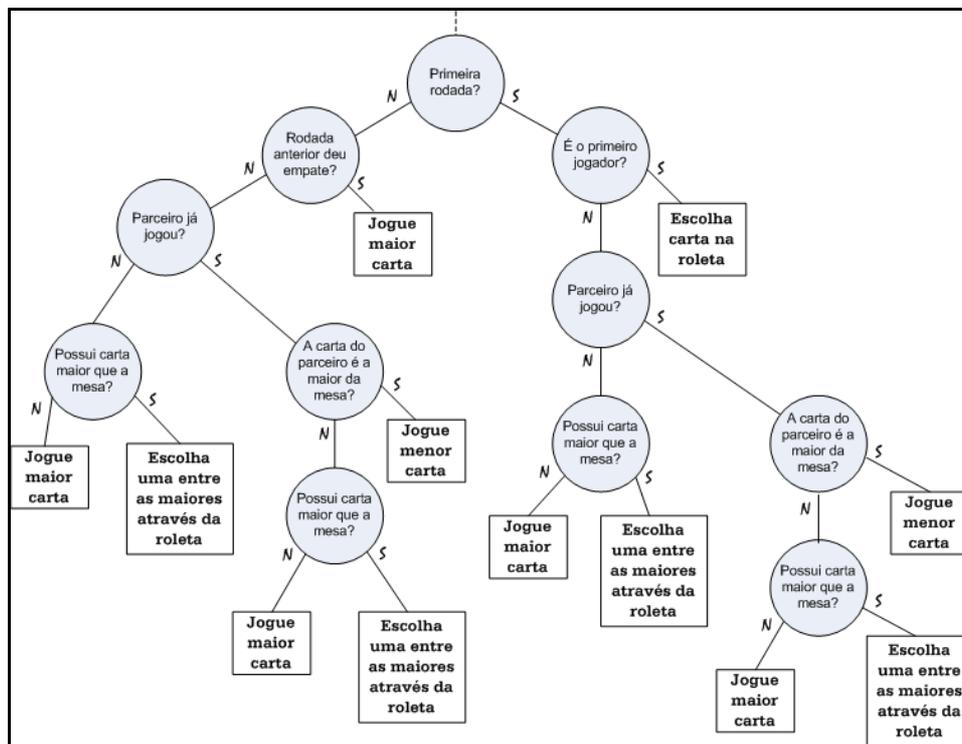


Figura 2. Rede semântica (Regras para Jogar uma Carta)

Com a definição das regras de jogo, foi possível mapear as interações necessárias entre os jogadores e as respectivas funcionalidades que foram desenvolvidas nos agentes. Como a comunicação entre os agentes é realizada através de um agente facilitador (Juiz), as trocas de mensagens são intermediadas por este agente. Na Figura 3 é apresentado um esquema da estrutura do protótipo, onde estão representadas as trocas de mensagens realizadas entre os agentes e as ferramentas utilizadas.

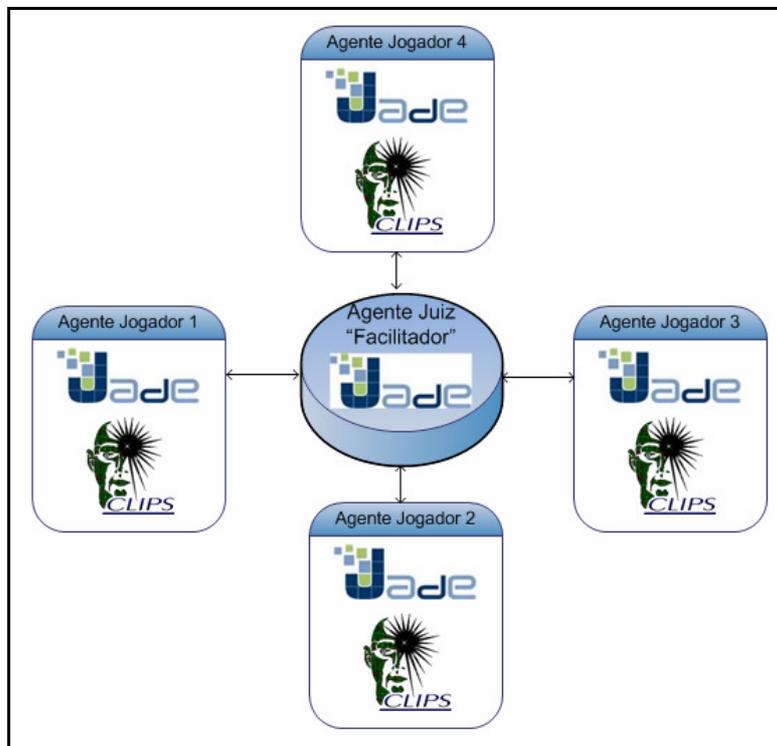


Figura 3. Esquema da estrutura do protótipo

Com a definição das regras e as trocas de mensagens foi realizada a implementação do protótipo utilizando a ferramenta JADE.

JADE funciona como uma “*thread*” que emprega múltiplas tarefas ou comportamentos e conversações simultâneas (SILVA, 2003). Esse agente Jade é implementado como uma classe Java chamada Agent. Essa classe Agent atua como uma super classe para a criação de agentes de software definidos por usuários. Ela provê métodos para executar tarefas básicas de agentes, tais como: passagens de mensagens, escalonamento, execução de múltiplas atividades concorrentes e interação simplificada com sistemas de agentes FIPA para a automação de tarefas comuns de agentes.

Desta forma, cada atividade que o agente executa, seja troca de mensagem ou ações como jogar uma carta, foi desenvolvida como um comportamento, onde o agente envia uma solicitação e recebe uma resposta do respectivo agente. Para a implementação das regras do Sistema Especialista, conforme apresentadas através das redes semânticas, foi utilizada o JClips.

Também foi elaborada uma interface visual para acompanhar o andamento do jogo, onde são apresentadas as cartas de cada rodada, o placar e demais informações do jogo. Na interface do agente Juiz (Figura 4) é apresentada a manilha e as cartas que cada jogador joga, bem como o placar da rodada e o placar geral. Também é visualizado o valor da rodada e o se algum jogador pediu truco.

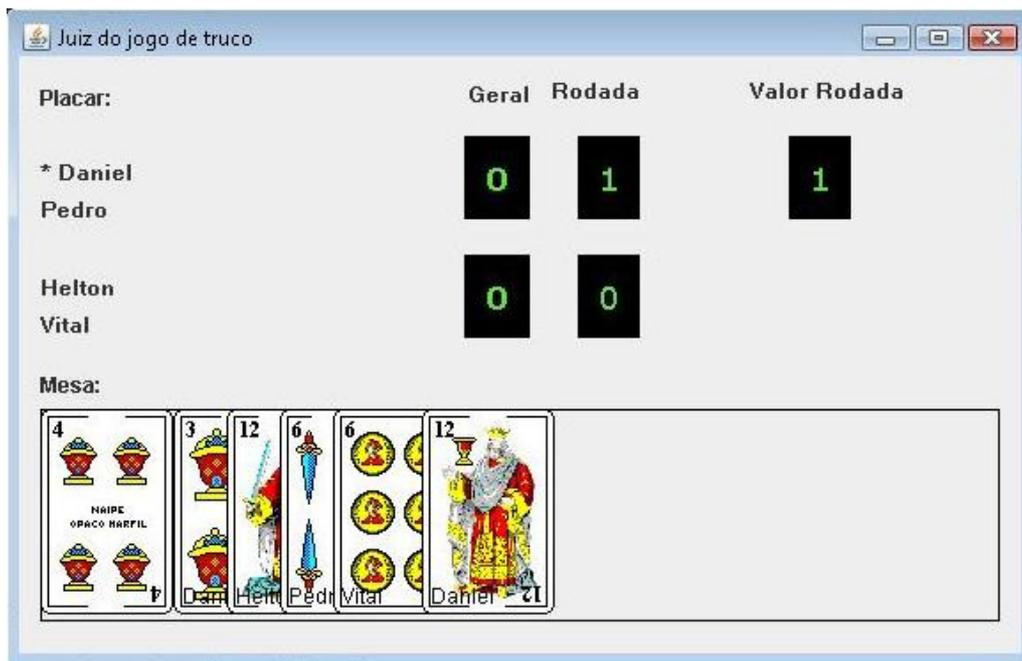


Figura 4. Interface do Agente Juiz

Na interface do agente Jogador (Figura 5) é apresentada a manilha, as cartas que o jogador recebe no início da rodada, o placar e as cartas que são jogadas pelos outros jogadores.

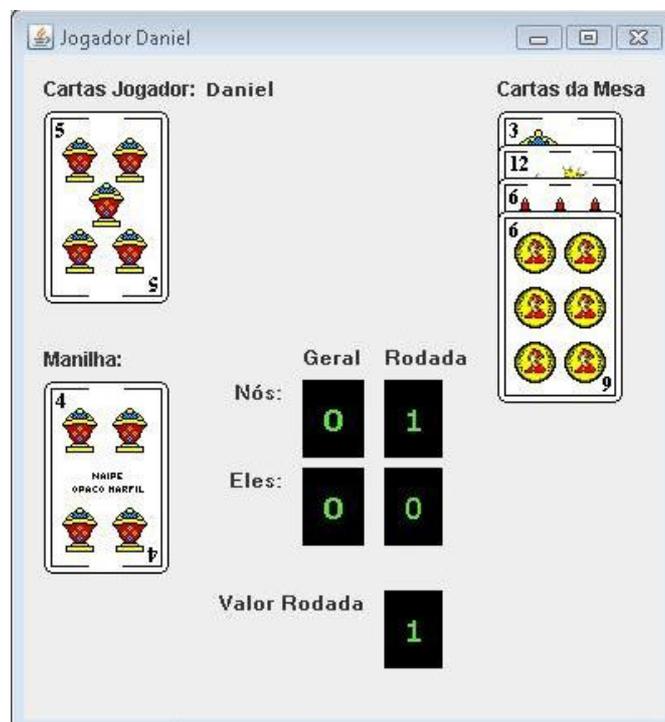


Figura 5. Interface do Agente Jogador

Para o início do jogo, primeiramente, é necessário iniciar o agente Juiz e em seguida os quatro jogadores. Cada jogador envia uma mensagem ao Juiz solicitando o seu registro. Quando o juiz identifica que estão registrados quatro jogadores, ele define as equipes e inicia automaticamente a partida. O jogo é finalizado quando uma equipe alcança o total geral de 12 pontos.

6. CONCLUSÃO

Pode-se constatar que sistemas multiagentes têm evoluído e mostrado sinais claros de que possuem um enorme potencial computacional. A aplicabilidade da tecnologia de agentes vem crescendo rapidamente o que deixa margem para surgimento de ambientes multiagentes variados, necessitando de uma padronização para manter um grau de interoperabilidade.

Após os experimentos realizados em sala de aula, ficou claro para o grupo de alunos, do papel das técnicas de inteligência artificial. Os estudos realizados sobre agentes inteligentes, somados aos aspectos de tomadas de decisão do sistema especialista, resultaram no desenvolvimento de um protótipo, o qual contribuiu de forma expressiva para a compreensão dos conceitos e sua aplicação prática.

A maior dificuldade encontrada foi a falta de material disponível relacionado as tecnologias utilizadas no trabalho, o que não foi um empecilho para o aprendizado, mas sim um desafio que mobilizou os alunos na busca do conhecimento.

Todavia, o sucesso foi alcançado principalmente devido a cooperação de todos os membros da equipe, que juntos definiram os agentes, as regras e a interface, tornando a experiência gratificante, sendo motivada pela essência do jogo de truço.

Como trabalhos futuros pretende-se aprimorar as regras do sistema especialista, possibilitando definir níveis de dificuldade para cada agente e também criar uma interface onde seja possível um jogador humano interagir com os agentes do sistema.

Esperamos com este trabalho ter trazido alguma contribuição no sentido de aprimorar e fornecer uma experiência prática do funcionamento de sistemas multiagentes, incorporando sistemas especialistas, agregando conceitos de diferentes técnicas através do estabelecimento de uma metodologia e da implementação de um protótipo, utilizando uma arquitetura funcional baseada em uma sociedade de agentes.

6. REFERÊNCIAS

BARRETO, J. M. “Inteligência Artificial no limiar do século XXI”, Florianópolis, O Autor. 2001.

BOCCA, E. W., JAQUES, P., VICARI, R. “Modelagem e Implementação da Interface para Apresentação de Comportamentos Animados e Emotivos de um Agente Pedagógico Animado”, In II Ciclo de Palestras de Informática da Educação, Porto Alegre/RS. 2003.

COPAG “Regras Oficiais de Jogos de Carta”, 7ª edição, Copag. 1990.

FERNANDES, A. M. R. “Inteligência Artificial”. Florianópolis: Visual Books. 2003.

JESS “JESS: the expert system Shell for the Java platform”. Disponível em: <http://www.jessrules.com/>. 2008.

LIA/UFC “Expert Sinta Shell”. Disponível em: <http://www.lia.ufc.br/>. 1996.

MENKEN, M. “JCLIPS: Clips for Java”. Disponível em: <http://sourceforge.net/projects/jclips/>. 2008.

MOURA, M. F. E CRUZ, S. A. B. “Estudo de expert system shells para o ambiente de diagnose remota”, Embrapa, Campinas. 2001.

RILEY, G. “CLIPS: A Tool for Building Expert Systems”. Disponível em: <http://clipsrules.sourceforge.net/>. 2008.

RUSSEL, S., NORVIG, P. “Inteligência Artificial”. Rio de Janeiro: Elsevier. 2004.

SILVA, L. A. M. “Estudo de Desenvolvimento de Sistemas Multiagentes usando Jade: Java Agent Development Framework”, Trabalho de Conclusão de Curso (Graduação em Ciências da Computação) – Centro Tecnológico. Universidade Federal de Pernambuco, Pernambuco. 2003.