UNIUBE
Educação e Responsabilidade social

VII ENTEC – Encontro de Tecnologia da UNIUBE
16 a 18 de novembro de 2011
Campus Aeroporto – Uberaba

INTELIGÊNCIA ARTIFICIAL PARA ROBOCUP SIMULATION LEAGUE 2D

Maciel, Vinícius Fonseca[1]; Silva, Pablo Henrique Penha[1]; Soares, Alexsandro Santos[2]; Lopes, Carlos Roberto[2]
[1] Universidade Federal de Uberlândia
[2] Faculdade de Computação - FACOM
viniciusfm@comp.ufu.br
alex@facom.ufu.br
crlopes@facom.ufu.br

1 - Introdução

O evento RoboCup, segundo a The RoboCup Federation, é uma competição internacional que visa o estudo e desenvolvimento de técnicas de Inteligência Artificial para robótica. Dentre suas modalidades destaca-se a Simulation League 2D, na qual se desenvolve um time de futebol baseado em simulação de robôs autônomos.

UFUbots é o primeiro time criado pela Faculdade de Computacão da Universidade Federal de Uberlândia para a participação nesta modalidade. Sua implementação faz uso de lógica fuzzy tanto no aspecto estratégico quanto na parte tática. Um diferencial deste time em relação a outros times participantes das competições anteriores é o uso ativo de um algoritmo de aprendizagem automática de regras fuzzy baseado em algoritmos genéticos.

Temos como nosso objetivo na competição RoboCup, além de estimular outros grupos da própria universidade a criarem seus times, adquirir conhecimento sobre técnicas de Inteligência Artificial aplicados à sistemas multiagentes e em tempo real.

Este artigo descreve o time UFUbots 2011, assim como as técnicas utilizadas em sua construção.

2 – Materiais e métodos

No desenvolvimento do UFUbots, resolvemos basear a estratégia do time no agent2d-3.0.0 que utiliza a biblioteca librcsc-4.0.0 e oferece os comandos básicos de comunicação com o simulador da RoboCup. Ele também oferece uma vasta quantidade de habilidades e define uma estratégia simples para a equipe. Este framework é de código aberto, implementado em C++, possui uma documentação para a simulação 2D da RoboCup e apresenta uma estrutura que facilita a modelagem do comportamento de cada jogador. As técnicas utilizadas para o time dividem-se em duas partes: estratégia (concentrada no técnico) e tática (realizadas pelos jogadores). Utilizamos dois sistemas baseados em regras fuzzy para implementar estas duas partes. As regras estratégicas possibilitarão ao time assumir diversas formações ao longo do jogo. Um exemplo é a formação defensiva (4-4-2), na qual os jogadores tentam impedir o avanço do time adversário, concentrando oito jogadores no seu campo e somente dois no campo adversário. Tal regra poderia ser enunciada como:

Se (posição da bola é muito perto) e (tempo da partida está no começo) então (formação é 4-4-2)..

As regras táticas possibilitam a cada jogador agir de acordo com uma dada situação. Com base na sua posição no campo, energia, número de jogadores da equipe entre o jogador e o gol adversário, distância do jogador adversário mais próximo que esteja entre o jogador e o gol adversário, o sistema fuzzy decide pelos seguintes comportamentos: manter a bola, driblar, passar a bola e chutar para o gol. Um exemplo de tal regra é:

Se jogador (energia) está muito cansado então (comportamento é) passar a bola.

Usamos o programa xFuzzy que nos auxiliou na criação e desenvolvimento das regras fuzzy. Com este programa, segundo Barros, temos a visão completa de como o sistema e as regras fuzzy funcionarão sem termos que, necessariamente, implementá-las no nosso time. Como exemplo de uso do

programa xFuzzy, temos a Figura 1 abaixo, onde esta mostra o gráfico gerado a partir dos tipos e valores de entrada do sistema fuzzy construído para modelar a força do chute de um jogador.

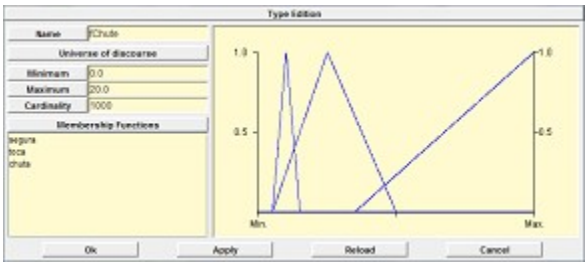


Figura 1 – Gráfico gerado pelo xFuzzy.

Um sistema baseado em lógica fuzzy é composto por uma base de conhecimentos (BC) e um mecanismo de inferência (MI). A base de conhecimentos é constituída por um conjunto de regras fuzzy, que são regras capazes de expressar ambiguidade e subjetividade presentes no raciocínio humano.

O mecanismo de inferência é capaz de manipular as regras da BC utilizando inferências e desta forma derivar uma saída ou conclusão de interesse. É comum que as regras para a especificação de cada comportamento do jogador sejam desenvolvidas manualmente. Entendemos que este processo poder ser melhorado para que se tenha um time com maiores possibilidades de sucesso. Portanto, na nossa implementação, decidimos estudar uma forma de gerar automaticamente tais regras. Desse modo, estamos estudando várias abordagens para serem usadas na geração automática de bases de conhecimento. Dentre as técnicas mais bem sucedidas estão os algoritmos de agrupamento (*clustering*), os métodos baseados em gradientes, as redes neurais e os algoritmos genéticos, segundo Cintra.

Recentemente tem havido um esforço considerável de pesquisa no sentido de se usar algoritmos genéticos (AG) na geração de sistemas fuzzy, em particular na geração das regras fuzzy com base em conjuntos fuzzy previamente definidos e fixos. Dado que o espaço de busca do AG é formado por todas as regras possíveis faz-se necessário o uso de algum mecanismo, em geral heurísticas, para facilitar definição e codificação dos cromossomos e aplicação de operadores genéticos.

Neste trabalho adotaremos o uso de algoritmos genéticos (AG) com o objetivo de gerar o conjunto de regras fuzzy com base em conjuntos fuzzy previamente definidos e fixos, tais como no trabalho, segundo Cintra, onde são definidos a heurística, a forma de codificação e tamanho dos cromossomos bem como do cálculo do valor de aptidão. No trabalho citado os experimentos tiveram 250 iterações, taxa de elitismo de 5%, taxa de cruzamento de 70% e taxa de mutação de 5% obtendo resultados significativos.

Na população inicial do AG introduzimos as regras descritas por Fraccaroli, Carlson (2010) e Batista (2010). Todas as demais regras foram geradas aleatoriamente. Na época da escrita deste artigo ainda estamos aprendendo as técnicas para a implementação do sistema de aprendizagem. Entretanto, para testarmos a viabilidade da utilização de regras fuzzy, codificamos manualmente as regras para o técnico e para os jogadores. Esta codificação inicial servirá como base de comparação para as regras aprendidas. Esperamos com isto demonstrar a viabilidade da utilização de algoritmos de aprendizagem no domínio da simulação 2D do RoboCup.

Para a implementação do time utilizaremos a linguagem de programação funcional Bigloo Scheme (www-sop.inria.fr/mimosa/fp/Bigloo), por ser uma linguagem considerada rápida e também pela facilidade de comunicação com a linguagem C++, esta usada na implementação do agent2D.

3 – Resultados e discussão

Os resultados que já obtivemos baseiam-se no aprendizado e uso da lógica fuzzy com a finalidade de criar bases de conhecimentos para os robôs autônomos. Já conseguimos desenvolver toda a parte de comunicação do nosso time.

Estamos agora trabalhando nas estratégias de time e incorporação das regras fuzzy nestas. Estamos também traduzindo a documentação do Agent2D (de japonês para português) e pesquisando sobre os algoritmos genéticos.

4 – Considerações finais

Fizemos uma análise da competição do ano passado nos times que competiram na RoboCup e observamos que a maioria deles utilizaram, com sucesso, a lógica fuzzy. Por isso, resolvemos também experimentar esta técnica. Notamos também que não foram citados algoritmos de aprendizagem o que nos

levou a considerar a implementação de um AG, com a finalidade de aprender regras úteis tanto para a estratégia quanto para a tática. Embora ainda estejamos em um estágio preliminar do nosso experimento, acreditamos que a nossa abordagem poderá mostrar-se vantajosa em longo prazo, quando os times participantes estiverem mais complexos e as regras fuzzy neles utilizadas forem mais difíceis de serem elaboradas manualmente.

5 - Agradecimentos

Os autores são bolsistas do Programa de Educação Tutorial / MEC.
Agradecemos também ao tutor Autran Macêdo pelo incentivo da escrita deste documento.

6 – Referências

The RoboCup Federation. Description of the RoboCup Simulation League. Disponível em: www.robocup.org Acesso em: 18 out. 2011.

BATISTA, Murillo R. et al. Descrição da Equipe USPDroids-2D, USP São Carlos, 2010. Disponível em: http://www.larc10.fei.edu.br/TDPs/2D/77908.pdf Acesso em: 8 maio 2011.

CINTRA, Marcos Evandro; CAMARGO, Heloisa de Arruda. Geração de Regras Fuzzy com Pré-seleção de Regras Candidatas. Universidade Federal de São Carlos, 2007. Disponível em: http://www.dcc.fc.up.pt/~ines/enia07_html/pdf/27966.pdf Acesso em: 5 maio 2011.

FRACCAROLI, Eduardo S.; CARLSON, Pedro M. Time GEARSIM 2010 da categoria RoboCup Simulation 2D. Universidade de São Paulo, 2010. Disponível em: http://www.cbr10.fei.edu.br/TDPs/2D/78444.pdf Acesso em: 8 maio 2011.

BARROS, Angel Barriga. et al. XFuzzy 3.0 description, 1997. Disponível em: http://www2.imse-cnm.csic.es/Xfuzzy/index.html Acesso em: 8 ago. 2011.

MONTESINO, F. et al. Fuzzy End-to-End Rate Control for Internet Trasnport Protocols. Vancouver, 2006. Disponível em: http://www.imse-cnm.csic.es/online/2006/F UZZIEEE2006.FMP.pdf Acessado em: 9 maio 2011.

GEORGE, Klir J. Fuzzy Thinking: The New Science of Fuzzy Logic, United States of America: Hyperion, 1994.