Generalização da aprendizagem por reforço: Uma estratégia para robôs autônomos cooperativos

Samir E. H. Kerbage, Emerson O. Antunes, Diego F. Almeida,
Paulo F. F. Rosa

***Resumo*—Este artigo descreve o time RobotIME 2D, desenvolvido no Instituto Militar de Engenharia. Serão apresentados as principais implementações da equipe que é baseada em aprendizado por reforço.**

I. INTRODUÇÃO

O interesse de pesquisa da comunidade de Inteligência Artificial em Sistemas Multiagentes tem gerado o crescimento da utilização de técnicas de agentes nas mais diversas aplicações em que esses sistemas podem ser usados, como por exemplo: jogos de computadores, interfaces adaptativas, simulação e controle de processos industriais.

A utilização do ambiente da RoboCup 2D para simulação de uma partida de futebol (simulador *Soccer Server*) permite a avaliação de diferentes técnicas de Sistemas Multiagentes como planejamento de estratégias, geração de conhecimento em tempo real, colaboração de agentes, princípios de agentes autônomos, entre outros, e estimula as pesquisas, investigações e testes que possibilitem a construção gradativa de agentes avançados.

Um pré-requisito para participação em uma competição na modalidade de simulação 2D de futebol é a disponibilização do código fonte do time participante. Portanto, este artigo foi desenvolvido a partir do código fonte do time UvA Trilearn [BOER; KOK, 2002], adicionando a seu código implementações das técnicas de Aprendizado por Reforço. Quando se deseja solucionar uma variedade de problemas e quando não existem modelos disponíveis a priori, o Aprendizado por Reforço é uma técnica muito atraente, pois o agente irá aprender a cumprir uma função de maneira correta em um ambiente desconhecido através de tentativa e erro.

No Aprendizado por Reforço, o agente aprende por meio da interação direta entre o agente e o ambiente, e das recompensas recebidas. Estas recompensas são dadas na forma de reforços positivos e negativos, que são usados para sinalizar ao agente se o mesmo está realizando as ações corretas ou não.

A tarefa de aprendizagem por reforço consiste em usar recompensas observadas para aprender uma política ótima (ou quase ótima) para o ambiente. Em uma plataforma de robôs autônomos cooperativos (e.g., o futebol de robôs), o programa pode ser informado quando ganhou ou perdeu e pode usar essa informação para aprender uma função de avaliação que forneça estimativas razoavelmente precisas da probabilidade de ganhar a partir de qualquer posição dada.

Samir Elias Hachem Kerbage, Seção de Engenharia de Computação samirehk@gmail.com
Emerson de Oliveira Antunes, Seção de Engenharia de Computação emersono10@hotmail.com
Diego Felix de Almeida, Seção de Engenharia de Computação diegobill@gmail.com
Paulo Fernando Ferreira Rosa, Seção de Engenharia de Computação rpaulo@ime.eb.br

II. MODELAGEM DA ESTRATÉGIA PARA O TIME

A implementação do algoritmo de Aprendizado por Reforço necessita de discretização do ambiente a fim de obter um número de estados viável para aplicação prática, devido às limitações de memória e principalmente de tempo.

Foi criada uma chave de estados que contém informações da localização do agente; do ângulo entre o adversário mais próximo, da bola e do agente; da distância do agente mais próximo; da distância do adversário mais próximo e por final da posição do adversário mais próximo(à direita, em frente ou à esquerda do agente). Estes valores são adicionados à chave por uma representação qualitativa, discretizados em valores que vão de zero a cinco. Com esta chave trabalharemos com 6x3x6x6x3 = 1944 estados. Cada estado constitui uma informação relevante para que o agente decida entre uma das cinco possíveis ações disponíveis.

A. Discretização dos estados

i0	i1	i2	i3	i4
[0;5]	[0;2]	[0;5]	[0;5]	[0;2]
Posição do agente no campo	Ângulo entre o oponente mais próximo, a bola e o aliado mais próximo	Distância do agente ao jogador do mesmo time mais próximo	Distância do agente ao jogador oponente mais próximo	Posição do oponente mais próximo em relação ao agente

Tabela I
Chave de estados

A posição do agente no campo (i0) é informada com o valor dado como 0 se o agente está localizado na área central da defesa, como 1 se o agente está em qualquer outra posição dentro da área da defesa, como 2 se ele está localizado na área central do meio de campo, como 3 se está localizado em outra posição dentro do meio do campo, como 4 se estiver localizado na área central do ataque e como 5 se estiver em outra posição dentro da área de

ataque. A Figura 1 exemplifica o que foi descrito. Esta discretização do campo valoriza a do UvA Trilearn de posicionar os jogadores mais velozes nas posições laterais de sua formação 4-3-3.

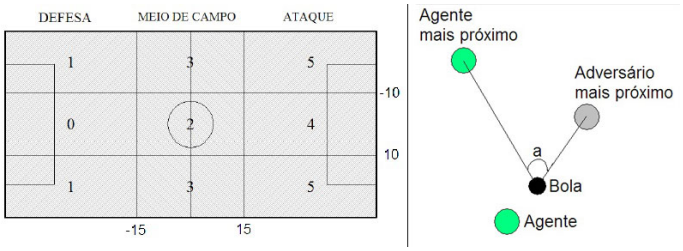

Figura 1. Representação da localização do índice (i0) da chave de estados e determinação do índice (i1) da chave de estados.

Os ângulos de i1 são representados na tabela II.

Ângulo (em graus)	Representação
$\|a\| <= 15$	0
$15 < \|a\| <= 30$	1
$30 < \|a\|$	2

Tabela II
Representação dos ângulos

As distâncias de (i2) e (i3), são representadas na tabela III.

Distância	Representação
$\|d\| <= 5.0$	0
$5.0 < \|d\| <= 10.0$	1
$10.0 < \|a\| <= 20.0$	2
$20.0 < \|a\| <= 35.0$	3
$35.0 < \|a\| <= 50.0$	4
$50.0 < \|a\|$	5

Tabela III
Representação das distâncias

B. Discretização das ações

Há uma enorme quantidade de ações realizáveis por um agente da RoboCup. Para a aplicação do algoritmo foi necessário reduzir o número de ações realizáveis, mas sem incapacitar o jogador. Procurou-se dar uma ênfase maior em ações para adoção de uma estratégia ofensiva. Outro ponto a se destacar é que o aprendizado só ocorre quando o agente está de posse da bola. Ações com a bola:

- Chutar para o gol adversário;
- Driblar na direção 30˚;
- Driblar na direção 0˚;
- Driblar na direção -30˚;
- Tocar para o jogador mais próximo.

C. Definição dos reforços

Reforços com avanço em direção ao gol adversário:

- Gol feito: +50;
- Posse da bola: +10;
- Adversário toma a bola: -10;
- Adversário marca gol: -100.

Reforços sem avanço em direção ao gol adversário:

- Posse da bola: +5;
- Adversário toma a bola: -20;
- Adversário marca gol: -100.

Na próxima seçã é feita uma descrição de como foi feita a implementação da estratégia de aprendizagem por reforço descrita nesta seção em um time de futebol de robôs da liga simulada 2D utilizando o UvA Trilearn como time base.

III. IMPLEMENTAÇÃO DA ESTRATÉGIA

Para realização da estratégia descrita na seção II foram feitas algumas adaptações no time base original do UvA Trilearn com as seguintes alterações:

- Encontrar oponente mais próximo e retornar sua posição;
- Encontrar jogador do mesmo time mais próximo e retornar sua posição;
- Retornar distância do agente à bola;
- Retornar distância do jogador do mesmo time mais próximo;
- Retornar distância do oponente mais próximo.

Além disso, foram implementadas as classes descritas a seguir.

A. Classe Matriz

Nesta classe foram concentrados os métodos para manipulação e armazenamento do conhecimento adquirido durante os jogos. No início de uma partida o agente instancia uma Matriz e a carrega com as experiências que estão armazenadas num arquivo comum para toda a equipe. Esta matriz é atualizada pelo agente que ao final da partida a armazena num arquivo referente a este agente. Como numa partida estão envolvidos 11 agentes do time em questão, são salvos 11 arquivos contendo as experiências obtidas por cada um dos agentes durante uma partida completa. Há, portanto a necessidade de centralizar em um único arquivo todo o conhecimento colhido. Isto é feito por uma classe auxiliar que tem o trabalho de ler os 11 arquivos referentes a cada agente e calcular a média do valor de cada posição da matriz de estados e salvar esse resultado em um único arquivo, que é o que será carregado pelos agentes numa próxima partida. Outro detalhe desta implementação é a necessidade de sincronismo dos processos, já que cada agente é na verdade um processo sendo executado separadamente, não havendo, portanto, uma ordem específica para que cada um termine sua operação. A solução adotada foi controlar o término dos processos através de um arquivo cujo código foi editado em *shellscript*.

B. Classe Crítico

Esta classe implementa o algoritimo Q-Learning de aprendizado por reforço. Neste método, para cada ação do agente é computado uma recompensa e o valor esperado ao seguir a melhor política, que é aprendida por meio da interação com o ambiente e, assim, aprendidos quais as melhores para chegar a um objetivo. É nesta classe ,portanto, onde ocorre a atualização da Matriz de estados, onde é calculado o novo valor para esta atualização com base nos valores de reforços e parâmetros do algoritmo. Também é onde o agente registra sua ação atual para uma posterior análise de seus efeitos.

Desta forma, a nova arquitetura do time é mostrada na figura 2.

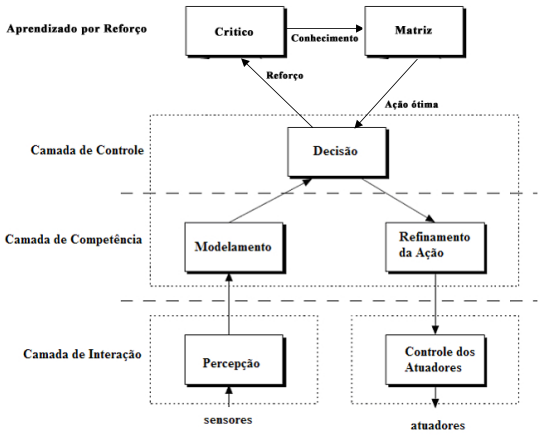

Figura 2. Arquitetura de camadas do time

A seguir será feita uma descrição de como foram feitos os treinamentos com o time implementado e análise dos resultados obtidos com esta abordagem.

IV. EXPERIMENTOS E RESULTADOS

Como no início dos treinamentos a Matriz de conhecimento não possui experiência alguma, optamos por iniciar os treinamentos com jogadas aleatórias e diminuindo a aleatoriedade à medida que a Matriz vai sendo preenchida. Adotamos a estratégia de treinamento abaixo utilizando o time UvA Trilearn como oponente, onde a taxa de exploração é o percentual de jogadas aleatórias.

- Diminuir em 10% a taxa de exploração a cada 30 jogos, iniciando no 100% até 0%;
- Executar 500 jogos com taxa de exploração de 0%

Foram realizados no total 800 treinos com o time implementado, onde observou-se os seguintes resultados:

Os resultados demonstraram uma convergência muito lenta, possíveis explicações para este resultado são:

- Falta de uma correta distribuição de reforços, e assim, os reforços não forneceriam a recompensa ideal pelas ações dos agentes e, deste modo, eles não aprenderiam com eficiência;
- Espaço de estados muito grande, se o agente precisa explorar uma grande área, ele demorará muito para fazê-lo e, quanto maior este espaço, maior será o tempo que o agente usará para a aprendizagem.
- O gráfico não apresentou um bom desempenho. Estes resultados demonstraram a necessidade de uma atenção maior na atribuição dos reforços, na determinação dos estados como subsídios para a tomada das decisões do agente e da importância de utilizar heurísticas para a aceleração do aprendizado.

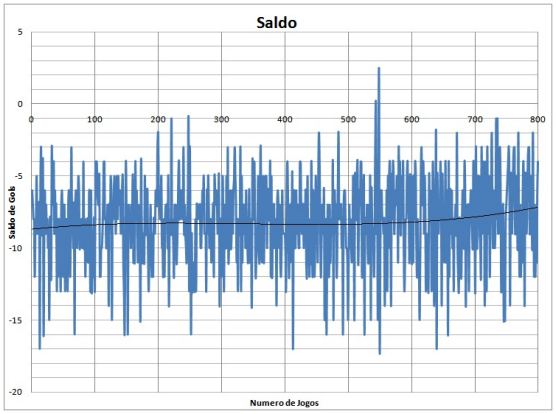

Figura 3. Gráfico da evolução do saldo de gols ao longo dos 800 treinos.

Concluimos pelos resultados dos treinamentos que o método de aprendizado por reforço, apesar de ser muito utilizado em ambientes desestruturados, quando é aplicado isoladamente, sem nenhuma heurística para aceleração da aprendizagem, possui convergência bastante lenta, indicando a necessidade de definição de algumas diretrizes adicionais para a aplicação do uso da aprendizagem por reforço no domínio do futebol de robôs, como o uso de heurísticas para acelerar o aprendizado e implementação de estratégias diferentes de acordo com a função que cada jogador assume, como as de atacante, zagueiro, meio de campo e goleiro.

V. CONCLUSÃO

Entre as principais vantagens da utilização de aprendizado por reforço no futebol de robôs que concluímos neste trabalho, podemos destacar:

- Agentes que podem se adaptar ao jogo. Os jogadores com aprendizado podem mudar suas características durante o jogo, caso o time adversário jogue , ou se o time adversário manter as mesmas características, os agentes podem se tornar um jogador melhor com o passar do tempo.
- Formas de discretização, trabalhando com uma discretização que ajudou no aprendizado em grandes espaços, sendo este um dos piores problemas para a elaboração de uma estratégia.

Como principais desvantagens encontradas durante este trabalho, destacam-se:

- O jogador aprende com as experiências a melhorar suas habilidades, mas ainda não se torna um jogador de destaque. Sendo assim, por exemplo, um atacante aprende com o passar do tempo a chutar para o gol, mas ainda não se torna o melhor atacante em campo. Novas melhorias precisam ainda ser estudadas.
- Times que alteram seu formato de jogar, tendem a atrapalhar o aprendizado. Ao aprender como o time joga, os agentes têm um comportamento, que pode mudar se o time adversário mudar também, mas esta mudança é gradual. Para times que mudam muito rapidamente, os agentes com aprendizado podem ter grandes dificuldades em aprender.

Para resolver algumas das desvantagens e para melhorar os resultados do trabalho, propõe-se para os trabalhos futuros:

- Estudar melhor as generalizações, com base nos resultados.
- Aplicar uma discretização maior das partes do ambiente que precisam de mais atenção, e uma menor para os que não precisam;
- Trabalhar com outros algoritmos de Aprendizado por Reforço;
- Utilizar novas formas de redução do tamanho de estados e assim poder levar os experimentos para o domínio dos multiagentes, fazendo trocas de aprendizado. O agente que aprender mais rápido ou que tiver a melhor eficiência em campo, ajuda os outros jogadores, passando o seu aprendizado;
- Usar o Aprendizado por Reforço acelerado por heurísticas no treinador, para otimizar a escolha de diversas táticas de jogos, e para melhor posicionar os jogadores que entram em campo, de acordo com suas características;
- Utilizar o Aprendizado por Reforço especializado para a função do jogador no campo. Por exemplo, o atancante deve utilizar estratégias diferentes daquelas utilizadas pelo zagueiro;
- Implementar o aprendizado para o goleiro.

VI. AGRADECIMENTOS

Os autores do artigo agradecem a contribuição do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) através do Programa Institucional de Bolsas de Iniciação em Desenvolvimento Tecnológico e Inovação (PIBITI).

Referências

[1] D. D. d. S. BAGATINI, "Um sistema multiagente para o simulador soccerserver," Master's thesis, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2001.

[2] J. BOER, Remco de; KOK, "The incremental development of a synthetic multi-agente system: The uva trilearn 2001 robocup soccer simulation team," Master's thesis, Faculty of Science, University of Amsterdam, Amsterdam, 2002.

[3] M. A. F. de Sousa, "Uma plataforma para cooperação autônoma de múltiplos robôs," Master's thesis, Instituto Militar de Engenharia, Rio de Janeiro, 2008.

[4] C. J. C. H. WATKINS, "Learning from delayed rewards," Ph.D. dissertation, Cambridge University, England, 1989.

[5] Y. X. CHEN, Mao, "Robocup soccer server users manual," http://sserver.sourceforge.net/, 2002, Último acesso em 26/06/2010.

[6] L. S. RIEDMILLER, Martin, "Reinforcement learning for robot soccer," http://portal.acm.org/citation.cfm?id=1569254/, 2009, Último acesso em 26/06/2010.

[7] N. P. RUSSEL, S.J., *Inteligência Artificial*, 2nd ed. Rio de Janeiro: Elsevier, 2004, tradutor: SOUZA, V.D.

[8] L. A. C. JUNIOR, "Aprendizado por reforço acelerado por heurísticas no domínio do futebol de robôs simulado," Centro Universitário, FEI, São Bernardo do Campo, Trabalho de Conclusão de Curso, 2007.