DOI: 10.48011/asba.v2i1.1465

# Desenvolvimento de uma Interface Gráfica Didática para o Ensino de Aprendizado por Reforço com Futebol de Robôs

**Higor Santos de Jesus** * **André Luiz C. Ottoni** *

* *Centro de Ciências Exatas e Tecnológicas (CETEC)*
*Universidade Federal do Recôncavo da Bahia (UFRB)*
*Cruz das Almas, Bahia, Brasil*
*(e-mails: higor-sj@hotmail.com e andre.ottoni@ufrb.edu.br)*

**Abstract:** This article presents a graphical interface for conducting Reinforcement Learning (RL) experiments applied to the Robot Soccer domain. The proposed learning environment allows the user to work with the main RL parameters, with: learning rate and discount factor. The results obtained, from a proposed experimental methodology, show that it is possible to carry out a comparative study of learning performance in robot soccer based on data generated by the graphic interface presented.

**Resumo**: Este artigo apresenta uma interface gráfica para a realização de experimentos de Aprendizado por Reforço (AR) aplicado ao domínio do futebol simulado. O ambiente de aprendizado proposto permite ao usuário trabalhar com os principais parâmetros do AR, condicionado à diferentes combinações, como: taxa de aprendizado e fator de desconto. Os resultados obtidos, a partir de uma metodologia experimental proposta, evidenciam que é possível realizar um estudo comparativo de rendimento do aprendizado no futebol de robôs a partir de dados gerados pela interface gráfica apresentada.

*Keywords:* Reinforcement Learning; Robot Soccer; Machine Learning; RoboCup; Graphical interface.

*Palavras-chaves:* Aprendizado por Reforço, Futebol de Robôs, Aprendizado de Máquina, RoboCup, Interface Gráfica.

## 1. INTRODUÇÃO

O avanço tecnológico fomenta a integração e aplicação da robótica em diversos setores da sociedade, buscando promover praticidade, facilidade e eficiência (Barrientos et al., 2007; Castro, 2008; Pêgo et al., 2015; de Souza et al., 2019; Pinto, 2019; Guarenti and Ribeiro, 2019; Ottoni et al., 2020b). Nesse aspecto, é possível caracterizar a robótica de acordo com seus ramos de atuação, tais como: autônoma, industrial, inteligente, educacional, móvel, dentre outras (Matarić, 2014; Goulart et al., 2015; Almeida et al., 2019; Garcia et al., 2019).

No contexto da robótica móvel, as aplicações vão desde ambientes domésticos até militares (Romero et al., 2014). Nesse sentido, para promover o desenvolvimento de tecnologias para robôs móveis, surge a necessidade de desafios de pesquisa, como é o caso do Futebol de Robôs promovido pela *RoboCup Federation* (Osawa et al., 1996; Kitano et al., 1997). Esse domínio possibilita trabalhar com a relação de agentes autônomos em ambientes, tomadas de decisão, utilização de sensores, atuadores, além de uma colaboração multiagente (Júnior et al., 2006; Silva and Simoes, 2009; Ottoni et al., 2015). Uma das categorias da *RoboCup* é a *Soccer Simulation 2D* (Simulação 2D), composta por duas equipes com onze agentes virtuais, dentro de um ambiente multiagente modelado em duas dimensões (de Boer and Kok, 2002; Ecar et al., 2012). A Simulação 2D é uma liga essencial para a aplicação de técnicas de Inteligência Artificial, das quais pode-se destacar, por exemplo: Redes Neurais (Faria et al., 2010; Jolly et al., 2007) e a Lógica Fuzzy (Li and Chen, 2015).

Um outro método com vasta aplicação na robótica e no ambiente simulado de futebol de robôs 2D é o Aprendizado por Reforço (AR) (Bianchi, 2004; Martins, 2007; Neri et al., 2012; Ottoni et al., 2015). Seu processo de aprendizagem, consiste em aplicar reforços (penalidades ou recompensas) ao agente, buscando otimizar seu comportamento (ações) nas diferentes situações do ambiente (estados) (Kaelbling et al., 1996; Sutton and Barto, 2018).

No futebol de robôs, o AR tem se destacado em diversas pesquisas, onde o campo do futebol se divide em variadas configurações de estados que em conjunto com as jogadas (ações), definem a melhor tática de jogo. Kerbage et al. (2010) define em sua abordagem uma discretização de estados em áreas de defesa, meio de campo e ataque para definir as estratégias de comportamento dos agentes. Já o trabalho de Fabro et al. (2013), compreende uma discretização dos estados com base no posicionamento dos jogadores em relação a bola e aos oponentes. Em outros trabalhos, heurísticas são aplicadas na busca por desempenho do algoritmo de AR (Celiberto Jr et al., 2012). No entanto, apesar da relevância do AR para o futebol

de robôs, a literatura ainda carece de ferramentas para a execução de algoritmos de aprendizado e a análise de jogos da Simulação 2D. De fato, o AR possui diversos parâmetros que podem influenciar no desempenho do aprendizado, como taxa de aprendizado, fator de desconto e $\epsilon - greedy$ (Schweighofer and Doya, 2003; Sutton and Barto, 2018; Ottoni et al., 2018).

Baseado nesse contexto, o objetivo deste trabalho é apresentar uma proposta de interface gráfica para a realização de experimentos de AR, no domínio do futebol de robôs simulado. Para isso, é implementado um relevante algoritmo de AR: *Q-learning* (Watkins and Dayan, 1992). Além disso, é demonstrada uma metodologia experimental com estudos de casos de parâmetros do AR para a Simulação 2D. Uma iniciativa similar pode ser observada no trabalho de Ottoni et al. (2020b), onde uma interface interativa é utilizada para o estudo de AR para instância de navegação de agentes e problemas de otimização combinatória (Problema do Caixeiro Viajante e da Mochila Multidimensional).

Este trabalho está dividido como segue: a Seção 2 apresenta uma abordagem conceitual para o AR e um contexto concernente ao futebol de robôs, com destaque para a categoria de Simulação 2D; a Seção 3 relata a construção e abordagem do ambiente de aprendizado; a Seção 4, por sua vez, apresenta a metodologia didática proposta para o ensino do AR utilizando o Futebol de Robôs Simulado; os experimentos e estudo de casos atrelados a metodologia, são descritos na Seção 5. Por fim, a Seção 6 destaca as contribuições do trabalho, seus resultados e direcionamento para novas versões.

## 2. FUNDAMENTAÇÃO TEÓRICA

### *2.1 Aprendizado por Reforço*

O conceito do AR, está diretamente ligado a teoria dos Processos de Decisão de Markov, do inglês Markov Decision Processes (MDP) (Sutton and Barto, 2018). Sua modelagem é expressa por meio da decisão de estados, ações, recompensas e transição de estados. Os estados e ações são mapeados de acordo com a política de decisão, onde se retorna uma recompensa quando uma ação é executada em um determinado estado e o estado futuro. Assim, o AR aprende a medida em que as ações vão sendo executadas dentro dos estados e os valores de recompensa vão sendo maximizados (Sutton and Barto, 2018; Puterman, 2014; Pellegrini and Wainer, 2007).

Para trabalhar com esses processos de decisão, o AR dispõe de algoritmos, tais como: *Q-learning* (Watkins and Dayan, 1992) e o SARSA (Sutton and Barto, 2018). O *Q-learning*, possui diversas aplicações e é usado para definir a ação à ser executada em cada estado do ambiente, se baseando na maximização da recompensa. É realizado um mapeamento das ações e estados, que definem a dimensão da tabela de aprendizado (número de estados x número de ações), denominada Q. Assim, cada par (estado, ação) possui um valor definido a partir das recompensas, representando uma solução ótima quando esses valores estiverem bem estimados. A Equação (1), representa a política de atualização da matriz Q, para os pares (estado, ação):

$$Q_{t+1} = Q_t(s,a) + \alpha[r + \gamma max_{a'} Q(s_{t+1},a_t) - Q_t(s,a)], \quad (1)$$

sendo $Q_{t+1}$ o valor atualizado após uma nova recompensa, $Q_t(s,a)$ é o valor atual da tabela para um par (s = estado, a = ação), $\alpha$ é a taxa de aprendizado, $r$ é a recompensa gerada após a execução da ação $a$ no estado $s$, $\gamma$ é o fator de desconto e $max_{a'} Q(s_{t+1},a_t)$ é o valor correspondente ao próximo estado e sua ação com maior valor. O Algoritmo 1 apresenta o *Q-learning*:

**Algoritmo 1** *Q-learning*

Inicialize $Q(s_t,a_t) = 0$ para todos os pares $(s_t,a_t)$
**while** *o critério de parada não é satisfeito* **do**
  Observe o estado $s_t$
  Selecione $a_t$ de acordo com a política $\epsilon$-*greedy*
  Execute a ação $a_t$
  Receba a recompensa $r(s_t,a_t)$
  Observe o estado $s_{t+1}$
  Atualize $Q(s_t,a_t)$ de acordo com a Equação (1)
  $s_t \leftarrow s_{t+1}$
**end**

Os parâmetros presentes no Algoritmo 1 são de extrema importância para se alcançar o bom desempenho esperado. Como supracitados na Equação (1) do *Q-learning*, tem-se: a taxa de aprendizado ($\alpha$), que é responsável por definir a atuação das novas atualizações, onde quanto mais próximo de 1 (um) maior será a alteração dos valores e quando mais próximo de 0 (zero) menor será a sobreposição das recompensas; o fator de desconto ($\gamma$) tem como objetivo maximizar a soma das recompensas, possuindo também uma variação entre 0 e 1; e a política de $\epsilon$-*greedy* (Algoritmo 1), que serve para controlar a aleatoriedade na escolha das ações, de modo a buscar o melhor desempenho do aprendizado, onde quanto mais próximo de 1, mais ações aleatórias serão definidas ao longo dos episódios de aprendizado (Sutton and Barto, 2018; Ottoni et al., 2018).

### *2.2 Futebol de Robôs*

O futebol de robôs teve sua primeira edição em 1997, chamada *RoboCup*, com o objetivo de promover pesquisa em Inteligência Artificial (IA) e robótica, por meio de agentes físicos e virtuais. Engloba um gama de tecnologias, tais como: design de agentes autônomos, colaboração multiagente, aquisição de estratégias, planejamento e raciocínio em tempo real, robótica inteligente, dentre outras. Além disso, existe o intuito de criar até a metade do século XXI, um time de robôs capaz de vencer a equipe campeã da copa do mundo da FIFA (Osawa et al., 1996; Kitano et al., 1997; Kitano, 1998).

A *RoboCup* se divide em categorias com características específicas, como: *Humanoid, Standart Platform, Middle Size, Small Size, Simulation 2D e Simulation 3D*. Dentro da categoria de Simulação 2D, cada time é composto por onze agentes virtuais, que operam em um ambiente simulado multiagente, proporcionando diversas pesquisas com abordagem para o aprendizado de máquina, IA e tomadas de decisão (de Boer and Kok, 2002; Henn et al., 2017; Prokopenko et al., 2018).

Seu funcionamento é baseado em três módulos: *Soccer Server*, Monitor e *LogPlayer*. O servidor (*Soccer Server*) é responsável por simular o ambiente das partidas e todos seus desdobramentos, como os times e seus jogadores, as jogadas, faltas, placar, penalidades, dentre outras características inerentes a uma partida de futebol. O monitor, serve basicamente para construir a interface de visualização para o usuário, onde são expostas todas as informações advindas do servidor, permitindo um contato direto com o placar e disposição dos jogadores em campo, por exemplo. Já o *LogPlayer*, permite a visualização das partidas após as mesmas ocorrerem, a partir de arquivos de registros, os quais podem ser lidos e interpretados (Chen et al., 2003). A Figura 1 apresentar o monitor do Simulador da RoboCup.

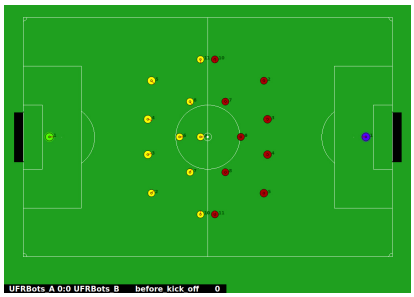


Figura 1. Monitor do Simulador 2D da *RoboCup*.

Como supracitado, o funcionamento principal dessa categoria se concentra no servidor, os outros módulos possuem atribuições destinadas à visualização. Os robôs virtuais (agentes), se conectam ao servidor através do protocolo UDP, como uma comunicação de três etapas: o agente envia as requisições ao servidor; o servidor armazena a requisição para executar (podendo ser chute, passe, drible, dentre outros); e o servidor devolve uma resposta atualizando o ambiente e as informações que o compõe (Chen et al., 2003).

## 3. AMBIENTE DE APRENDIZADO POR REFORÇO PROPOSTO

O ambiente para a metodologia proposta, foi construído com a linguagem de programação *Python*, para funcionar junto com o simulador da categoria de simulação 2D de futebol de robôs da *RoboCup*. Dessa maneira, se fez necessária a divisão do desenvolvimento em duas partes principais: (1) conexão e execução do simulador; (2) construção da interface que permite a interação do usuário com o sistema, permitindo a execução de partidas e aplicar na prática os conceitos do AR.

### *3.1 Modelagem da conexão com o simulador*

Esse módulo é fundamental para o funcionamento do ambiente de aprendizado e simulação das partidas do futebol de robôs. O ambiente de simulação 2D compreende em seu funcionamento, como já descrito, o servidor, o monitor e o *logplayer*. Para que sejam utilizados, se faz necessário o acesso a terminais no sistema GNU/Linux para a inserção de comandos do simulador. Por exemplo, para se iniciar uma partida, em um terminal se inicia o servidor, em outros dois os times que irão se enfrentar e em um quarto terminal é iniciado o monitor, quando desejado. Dessa maneira, o sistema deve executar todas as etapas exigidas pelo ambiente de simulação 2D. Com isso, o módulo de conexão e execução do simulador foi criado (Figura 2).

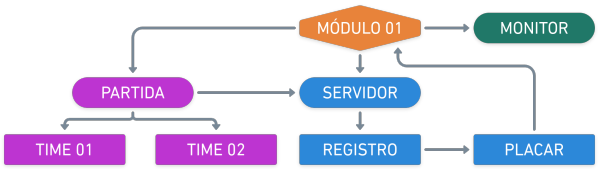


Figura 2. Módulo de conexão com o simulador.

Essa modelagem, permite a execução de partidas entre dois times quaisquer. Para isso, tem-se as seguintes funcionalidades na Figura 2:

- **Simulador**: módulo de conexão com o Simulador 2D, ou seja, responsável por executar as diretrizes necessárias para iniciar uma partida;
- **Servidor**: iniciar o servidor do simulador;
- **Partida**: componente responsável por preparar dois times para uma partida. Além de compilar o time de aprendizado, quando sofre alteração de seus parâmetros;
- **Time 01**: cria a instância do primeiro time selecionado;
- **Time 02**: cria a instância do segundo time selecionado;
- **Monitor**: inicia o monitor do simulador, quando requisitada sua exibição;
- **Placar**: retorna o placar, a partir da leitura do arquivo de registro gerado pela partida.

Em suma, esses componentes servem como base para o funcionamento para o ambiente de aprendizado, permitindo a execução das partidas/episódios junto ao Simulador 2D do futebol de robôs. Sua construção é responsável por acessar pastas/arquivos e executar comandos, não sensíveis ao usuário do sistema.

### *3.2 Modelagem da interface*

Essa seção descreve a construção da interface de interação com o usuário, a qual possui os módulos subsequentes (Figura 3):

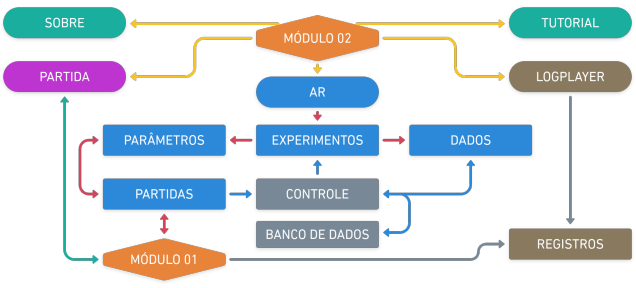


Figura 3. Módulo da interface gráfica.

- **Partida**: responsável por realizar um partida entre dois times que sejam escolhidos pelo usuário (linha verde). O módulo simulador (Figura 2) é chamado, e ao fim da partida retorna o placar à interface;

- **AR**: ambiente de aprendizado por reforço (linhas vermelhas), do qual compreende os seguintes aspectos:
  - Experimentos - realiza o controle dos experimentos que o usuário deseja executar. Permitindo o agrupamento dos aprendizados com diferentes diretrizes e finalidades;
  - Parâmetros - ao acessar um experimento, gerencia a definição dos parâmetros para a execução de um ensaio/aprendizado.
  - Partidas - realiza a comunicação direta com o módulo 01 (Figura 2), onde realiza a execução de todas as partidas necessárias para o aprendizado. Ao iniciar cada partida, é criada uma nova instância do módulo 01, que retorna o placar da partida, compondo dessa forma os resultados do aprendizado;
  - Controle - módulo responsável por trabalhar com a persistência dos dados gerados pelo ambiente de aprendizado (linhas azuis). Quando finalizado um ensaio, o módulo *Partidas* envia os resultados gerados e o mesmo armazena no banco de dados do sistema. Além disso, gerencia a disposição dos experimentos no módulo *Experimentos*, e retorna os dados requiridos pelo próximo módulo *Dados*;
  - Dados - ao acessar um experimento, esse módulo gerencia os resultados gerados durante os ensaios associados ao mesmo. Permite a consulta e análise dos dados por meio de tabelas e gráficos;
  - Banco de dados - armazena os experimentos, ensaios e parâmetros existentes no sistema.
- **LogPlayer**: disponibiliza ao usuário, a opção de assistir partidas já realizadas, por meio do módulo *LogPlayer* do Simulador 2D, a medida que consulta os registros das mesmas (linhas cinzas);
- **Tutorial**: exibe tutoriais do sistema, concernentes as funcionalidades principais;
- **Sobre**: apresenta informações relevantes ao sistema e aos envolvidos em sua construção.

Os itens citados acima compõe a interface inicial do sistema da Figura 4.


Figura 4. Página inicial do sistema de Aprendizado por Reforço, com acesso direto aos principais módulos.

O módulo partida (Figura 5), permite o usuário realizar partidas entre dois times quaisquer, necessitando apenas navegar até o diretório desejado, a partir dos botões *Time 1* e *Time 2*. Além disso, as configurações da partida podem ser alteradas: o modo normal corresponde a um tempo de partida de aproximadamente dez minutos, podendo ser alterado para o modo rápido que possui duração em torno de três minutos. Bem como pode-se escolher da mesma maneira, a exibição ou não do monitor, apenas clicando no botão subsequente. O botão *Info*, no canto inferior direito, exemplifica o passo a passo para a escolha dos times. O botão de iniciar a partida, só é exibido após seleção dos times, onde ao fim da mesma, o resultado é disposto na interface, informando o vencedor ou a situação de empate.

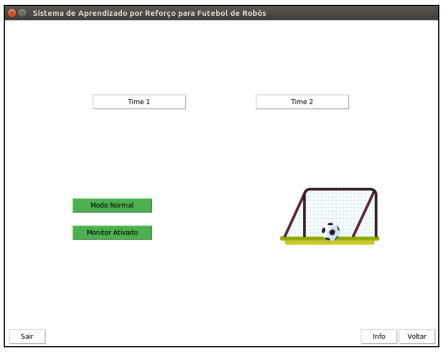


Figura 5. Módulo partida - destinado a realização de uma jogo entre dois times quaisquer.

O módulo aprendizado, é o que compreende o principal objetivo desse trabalho, a partir dele é possível o usuário aplicar os conceitos do AR. Sua utilização é composta em quatro telas (Figuras 6, 7, 8 e 9). A primeira tela (Figura 6), é responsável pelo controle dos experimentos. É disposta a opção de cadastrar um experimento, definindo um nome de descrição para o mesmo. Ao selecionar uma opção (experimento), é possível prosseguir para a definição de parâmetros e realizar um ensaio (botão *Next*), acessar os dados gerados por ensaios já realizados (botão *Dados*) ou realizar a remoção do mesmo.


Figura 6. Módulo de aprendizado - controle dos experimentos (cadastro, remoção e seleção).

Acessando a opção *Dados*, é exibida a tela da Figura 7. Em sua parte superior contém uma tabela, com as combinações de parâmetros criadas pelo usuário na tela de definição de parâmetros (Figura 8), exibindo as informações: médias (dentre os ensaios de mesma combinação) de saldo de gols, gols sofridos e gols feitos, taxa de aprendizado, fator de desconto e política $\epsilon - greedy$, para cada combinação. Em seguida um botão, que realiza a plotagem das médias de cada combinação. A segunda tabela, apresenta o detalhamento de cada ensaio, onde podem ser excluídos ou acessados através de gráficos das somas de saldos de gols, gols feitos e sofridos, além de arquivos de texto com os placares das partidas, ao clicar no botão *Ver Ensaio*.

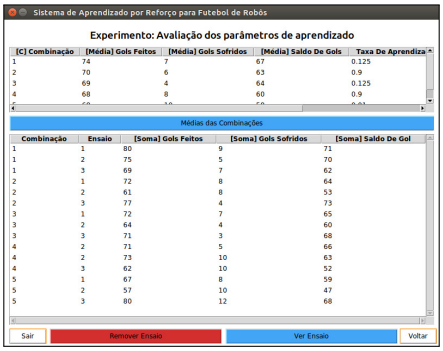


Figura 7. Módulo de aprendizado - acesso aos resultados dos ensaios realizados.

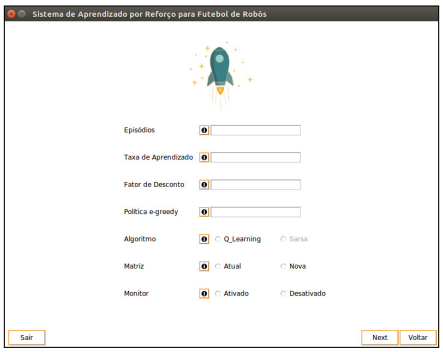


Figura 8. Módulo de aprendizado - definição de parâmetros.

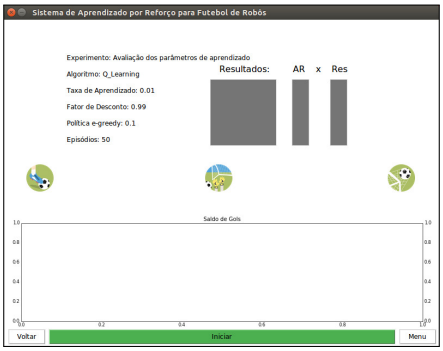


Figura 9. Módulo de aprendizado - execução das partidas.

A tela de parâmetros (Figura 8), recebe as configurações dos parâmetros de aprendizagem. Ao lado de cada opção de entrada, existe um botão de informação, que exibe uma pequena descrição dos mesmos. Segue a descrição dos parâmetros de entrada:

- **Episódios**: recebe o número de episódios que serão executados durante o aprendizado, ou seja, o número de partidas que o time do AR irá executar;
- **Taxa de Aprendizado**: parâmetro da equação de aprendizado ($\alpha$), descrito na Seção 2.1;
- **Fator de Desconto**: parâmetro da equação de aprendizado ($\gamma$), descrito na Seção 2.1;
- **Política** $\epsilon$ − *greedy*: política de aleatoriedade na escolha das ações, descrito na Seção 2.1;
- **Algoritmo**: seleção do algoritmo de aprendizado. Contudo, no momento da descrição desse trabalho, apenas o algoritmo *Q-learning* se encontra disponível;
- **Matriz**: se diz com respeito a matriz de aprendizado:
  - Atual - a matriz de aprendizado não será alterada. Essa opção é indicada para continuar o aprendizado do time, ou seja, se outros episódios já foram executados, o aprendizado poder ser continuado ao invés de reiniciar.
  - Nova - a matriz de aprendizado é inicializada com o valor 0 (zero), para todos pares (estado, ação). Opção indicada para iniciar o aprendizado com novos parâmetros ou atualização do time.
- **Monitor**: ativar ou não o monitor para visualizar as partidas, que são executadas no modo rápido (duração média de três minutos por partida).

A próxima tela (Figura 9), exibe os parâmetros do aprendizado, o resultado das partidas, um gráfico que é atualizado ao final de cada partida contendo o saldo de gols e uma barra de progresso, que se diz respeito a conclusão dos episódios/aprendizado.

O módulo de assistir partidas (Figura 10), é destinado especificamente para que as partidas do AR sejam assistidas posteriormente ao aprendizado. Acessando o botão do canto superior esquerdo, é possível navegar a outro diretório que possuam registros de partidas e assisti-los com o auxílio da interface. Basta selecionar uma opção da lista de registros exibida na interface e clicar no botão *Assistir*. Os registros das partidas ficam localizados na pasta *logAR* e das partidas fora do aprendizado na pasta *log*, ambas no diretório principal do sistema operacional.

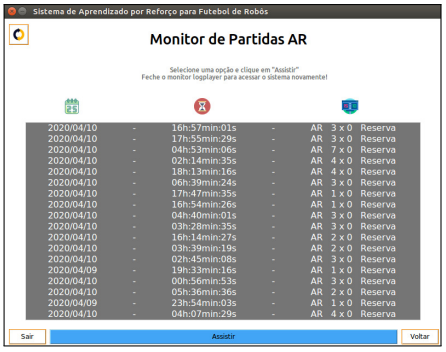


Figura 10. Interface destinada a assistir partidas realizadas.

### *3.3 Modelo de Aprendizado por Reforço*

O time de AR utilizado no sistema foi modelado em estados, ações e recompensas, baseando-se na estrutura apresentada em (Ottoni et al., 2015):

- **Ações**: chute ao gol; lançamento de bola para área ofensiva; passe normal; passe rápido; drible normal; e drible rápido.
- **Estados**: Foram definidos 32 estados. Para caracterizar os estados, o campo (Figura 1) é dividido em 8 zonas. Cada zona, possuem 4 estados/subdivisões, totalizando os 32 estados no ambiente. Para o mapeamento de cada estado para o time e tabela Q, são utilizadas as coordenadas X e Y do campo.
- **Reforços**: o time possui a diretriz principal de aprender a realizar gols. Dessa maneira, as recompensas foram definidas de modo a direcionar o time à realizar o chute ao gol, das quais recebem um aumento no seu valor a medida que o time avança para a zona

de ataque (estados dentro da área do gol adversário). Por exemplo, na zona 1 (defesa) a recompensa é de $-50$ e a penalidade de $-100$ e na zona 8 (ataque) a recompensa é de 50 e a penalidade de 25. O chute na zona de ataque, recebe uma recompensa de 500, valor maior em relação às outras ações.

Já o time reserva (adversário de treinamento), possui uma abordagem ofensiva, com lançamento de bola para área ofensiva por parte dos zagueiros, o meio de campo busca realizar o passe para os jogadores 9, 10 e 11, o jogador 10 dribla ou toca a bola para os jogadores 9 e 11, que são direcionados unicamente ao chute.

Os códigos dos times estão localizados no diretório *Documentos* do sistema operacional, sendo identificado pelo nome *AR_System*, bem como seu adversário (*Reserva _AR_System*). Vale ressaltar também que os dois times foram desenvolvidos a partir do código base do *UvA Trilearn* (de Boer and Kok, 2002).

## 4. METODOLOGIA EXPERIMENTAL

Esta seção apresenta a proposta de metodologia experimental didática para aplicação da interface gráfica desenvolvida. Sua execução é seguida por meio de experimentos práticos com o algoritmo *Q-learning* no domínio do ambiente simulado de futebol de robôs 2D. Através dos experimentos propostos é possível observar a influência da definição de parâmetros do AR nos resultados.

Em seguida, são apresentados os estudos de casos da metodologia proposta, onde o objetivo é identificar a influência provocada pela alteração nos parâmetros: taxa de aprendizado e fator de desconto. O estudo de caso 01 ($\alpha$ = 0,125 e $\gamma$ = 0,9) refere-se a uma combinação de parâmetros adotada com frequência em aplicações do AR no futebol de robôs (Celiberto Jr et al., 2012; Berton and Bianchi, 2015). Já outros estudos de casos, são variações das configurações de $\alpha$ e $\gamma$, a fim de ilustrar a capacidade da interface em mostrar influência experimental desses parâmetros no aprendizado.

**Estudo de caso 01** :
- Taxa de aprendizado ($\alpha$) = 0,125;
- Fator de desconto ($\gamma$) = 0,9;

**Estudo de caso 02** :
- Taxa de aprendizado ($\alpha$) = 0,9;
- Fator de desconto ($\gamma$) = 0,9;

**Estudo de caso 03** :
- Taxa de aprendizado ($\alpha$) = 0,125;
- Fator de desconto ($\gamma$) = 0,125;

**Estudo de caso 04** :
- Taxa de aprendizado ($\alpha$) = 0,01;
- Fator de desconto ($\gamma$) = 0,99;

Além dos parâmetros descritos, são fixados: a política $\epsilon - greedy$ em 0,1 e a execução de três ensaios (repetições) de 50 episódios para cada estudo de caso. Os ensaios podem ser executados seguindo as etapas da Figura 11. As setas de cor azul indicam o botão que deve ser clicado para avançar à próxima etapa, e as vermelhas indicam o andamento (condizente com a numeração). Segue a descrição dessas etapas:

**Passo 1** - Acesse através da interface principal o ambiente de aprendizado, selecionado o segundo botão *Aprendizado*;

**Passo 2** - Cadastre um experimento, definindo um nome para o mesmo, é através dele que serão consultados os resultados. Selecione o experimento e clique no botão **Next**;

**Passo 3** - Preencha os dados, de acordo com as informações dos estudos de casos, descritos anteriormente, iniciando pelo Estudo de caso 01;

**Passo 4** - Clique no botão **Iniciar** e aguarde o fim do aprendizado;

**Passo 5** - Clique no botão **Resultados** para gerar o gráfico de visualização do aprendizado, bem como os placares;
- Serão abertos três arquivos, o primeiro contendo a relação dos placares de cada partida, bem como outros dois com os resultados individuais, permitindo uma cópia dos mesmos, para possibilitar por exemplo a inserção em uma planilha externa para demais tratamentos. Também será gerado um gráfico com a plotagem do saldo de gols, gols do time AR (gols feitos) e gol do time adversário (gols sofridos);

**Passo 6** - Repita essas etapas até concluir todos os estudos de casos. Quando concluir, acesse através da tela dos experimentos, os resultados concernentes ao experimento cadastrado, clicando no botão **Dados** (seta amarela, Figura 11, passo 2).

Ao final de todos os experimentos, com o armazenamento dos registros, é possível realizar a comparação da influência dos parâmetros fator de desconto e taxa de aprendizado, nos resultados obtidos, no que se diz respeito a evolução do time AR. O módulo de visualização dos dados, disponibilizado pelo sistema, permiti o acessos às informações geradas por cada combinação de parâmetros.

## 5. RESULTADOS DOS ESTUDOS DE CASOS

Nesta seção são apresentadas os resultados das simulações dos estudos de casos. Vale ressaltar, que objetivo é mostrar que a interface gráfica possibilita identificar a influência de alteração nos parâmetros do AR no domínio do ambiente simulado de futebol de robôs 2D. A Seção está dividida em análise descritiva e análise de variância dos resultados.

### *5.1 Análise Descritiva*

A Figura 12, apresenta o gráfico (gerado pela interface) das médias de medidas de desempenhos (Gols Feitos, Gols Sofridos e Saldos de Gols) para cada estudo de caso.

Pela Figura 12 e Tabela 1, é possível perceber que a combinação 1 ($\alpha$ = 0,125 e $\gamma$ = 0,9) alcançou os melhores desempenhos em termos de médias de gols feitos (54,7) e saldos de gols (48,7). No entanto, em termos de saldos de gols no início do aprendizado (SGI), foi o estudo de caso 4 ($\alpha$ = 0,01 e $\gamma$ = 0,9) que alcançou os melhores resultados (7,7). Por fim, destaca-se os resultados da configuração 3 ($\alpha$ = 0,125 e $\gamma$ = 0,125) em termos de desempenho no fim de aprendizado, onde obteve média de SGF = 12,7, superando as demais configurações.

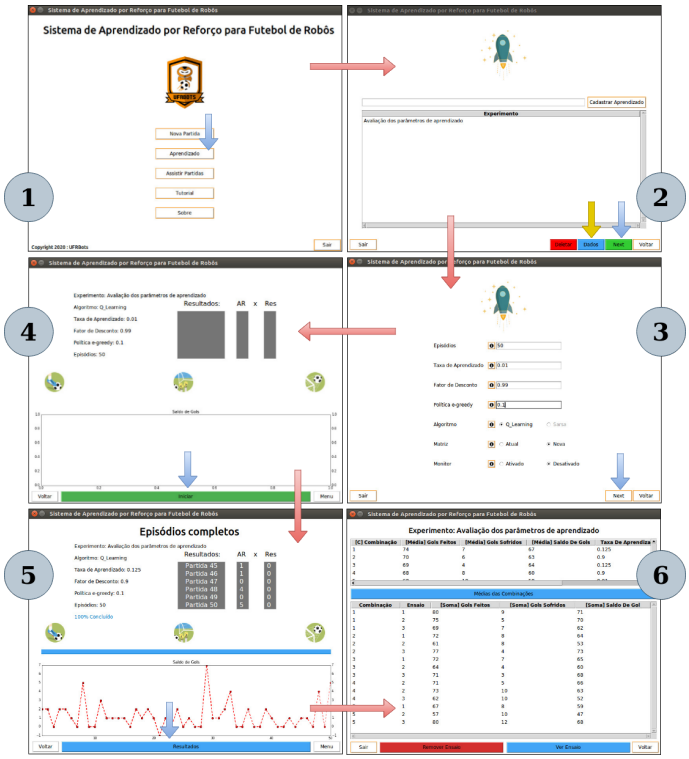


Figura 11. Etapas para realizar experimentos no ambiente de aprendizado por reforço.

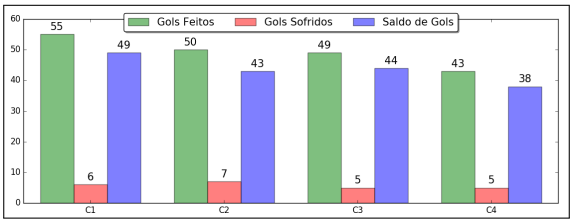


Figura 12. Médias dos estudos de casos

Tabela 1. Medidas de desempenho de GF (média de gols feitos); GS (média de gols sofridos); SG (média de saldo de gols); SGI (SG nos 10 episódios iniciais); e SGF (SG nos 10 episódios finais), para cada estudo de caso (EC). Em negrito, os melhores valores por medida de desempenho.

| *EC* | $\alpha$ | $\gamma$ | *GF* | *GS* | *SG* | *SGI* | *SGF* |
|---|---|---|---|---|---|---|---|
| 1 | 0,125 | 0,9 | **54,7** | 6,0 | **48,7** | 7,3 | 11,3 |
| 2 | 0,9 | 0,9 | 50,3 | 7,3 | 43,0 | 7,3 | 7,0 |
| 3 | 0,125 | 0,125 | 49,0 | **5,3** | 43,7 | 4,0 | **12,7** |
| 4 | 0,01 | 0,99 | 43,3 | **5,3** | 38,0 | **7,7** | 5,7 |

### *5.2 Análise de Variância*

Nesta seção, são apresentados experimentos de Análise de Variância (ANOVA) (Montgomery, 2017) para confirmar que existe diferença estatística significativa entre os desempenhos dos estudos de casos propostos. A ANOVA é uma técnica estatística que compara as médias dos tratamentos (estudos de casos) e já foi abordada na literatura como método de estimação de parâmetros do AR, como descrito em (Ottoni et al., 2020a). A medida de desempenho selecionada para ser analisada é o Saldo de Gols nos episódios finais de aprendizado (SGF). Isso porque, reflete o desempenho do time de robôs simulado após um número maior de partidas de treinamento.

O método estatístico de ANOVA avaliou se as médias populacionais dos resultados dos 4 estudos de casos ($\mu_1$, $\mu_2$, $\mu_3$, $\mu_4$) para SGF são estatisticamente iguais ou diferentes (Montgomery, 2017), através das hipóteses:

$$\begin{cases} H_0 : & \mu_1 = \mu_2 = \mu_3 = \mu_4, \\ H_1 : & \text{pelo menos um par de médias é diferente.} \end{cases}$$

A hipótese inicial ($H_0$) foi rejeitada e a hipótese alternativa ($H_1$) foi aceita, pois o resultado do teste foi p-valor = 0,0303 (adotando um nível de significância de 5%). Vale ressaltar que as medidas de adequação da ANOVA também foram observadas: normalidade dos resíduos, homogeneidade das variâncias e independência (Montgomery, 2017).

Nesse aspecto, após a análise dos resultados da ANOVA é possível concluir que ao menos uma das 4 configurações ($\alpha$ e $\gamma$) analisadas apresentou desempenho estatístico diferente em relação as demais em termos do SGF. Assim, reforça a relevância da interface gráfica proposta, pois possibilita realizar experimentos e avaliar como o desempenho pode ser influenciados por situações de aprendizado, como por exemplo, parâmetros.

## 6. CONCLUSÃO

O objetivo deste trabalho foi apresentar uma interface gráfica para realização de experimentos de AR, no domínio do futebol de robôs simulado. O ambiente de aprendizado proposto, permite o usuário trabalhar com os principais parâmetros do AR, condicionado à diferentes combinações, permitindo a avaliação do impacto das definições realizadas no rendimento do aprendizado, através de uma interface gráfica e disposição dos dados gerados.

Os resultados dos estudos de casos propostos, demonstram que é possível realizar o comparativo no rendimento do aprendizado, provocado pela combinação de parâmetros: taxa de aprendizado, fator de desconto, política $\epsilon - greedy$ e número de episódios. Além disso, evidencia que a metodologia proposta, pode auxiliar estudantes no aprendizado do AR.

Em trabalhos futuros, espera-se a evolução do sistema, bem como sua disponibilização para *download* e utilização. A inclusão de novos módulos, deve permitir outras opções de algoritmo de AR, como por exemplo o SARSA. Além disso, proporcionar, a definição de estados e ações nos parâmetros de aprendizagem, a definição de esquemas táticos para o time e a inclusão de outros métodos estatísticos que possam auxiliar na análise dos resultados das partidas e compreensão do aprendizado.

## AGRADECIMENTOS

Agradecemos à UFRB e UFSJ (Edital 001/2019/Reitoria).

## REFERÊNCIAS

Almeida, J.A., Cruz, M.E.J.K.d., Schulz, J.M., and Müller, V. (2019). Inclusão digital de crianças e jovens por meio da robótica educacional. *Anais do Salão de Ensino e de Extensão*, 230.

Barrientos, A., Peñin, L.F., Balaguer, C., and Aracil, R. (2007). *Fundamentos de robótica*, volume 2. McGraw-Hill Madrid.

Berton, P.A. and Bianchi, A.C. (2015). Aprendizado por reforço aplicado ao desenvolvimento de agentes humanoides no domínio do futebol de robôs simulado. *XII Simpósio Brasileiro de Automação Inteligente*.

Bianchi, R.A.C. (2004). *Uso de heurísticas para a aceleração do aprendizado por reforço*. Ph.D. thesis, USP.

Castro, V.G.d. (2008). Roboeduc: Especificação de um software educacional para ensino da robótica às crianças como uma ferramenta de inclusão digital. *PPGEE - Mestrado em Engenharia Elétrica e de Computação, UFRN*.

Celiberto Jr, L.A., Matsuura, J.P., De Màntaras, R.L., and Bianchi, R.A. (2012). Reinforcement learning with case-based heuristics for robocup soccer keepaway. In *2012 Brazilian Robotics Symposium and Latin American Robotics Symposium*, 7–13. IEEE.

Chen, M., Dorer, K., Foroughi, E., Heintz, F., Huang, Z.X., Kapetanakis, S., Kostiadis, K., Kummeneje, J., Murray, J., Noda, I., et al. (2003). *Users Manual: RoboCup Soccer Server, RoboCup Federation*.

de Boer, R. and Kok, J. (2002). The incremental development of a synthetic multi-agent system: The uva trilearn 2001 robotic soccer simulation team. *Master's thesis for Artificial Intelligence and Computer Science, University of Amsterdam*.

de Souza, A.H.G., Pinheiro, A.M., Moraes, A.C., dos Santos, D.M.G., Zago, G.M.P., de Oliveira, S.C., and Mendes, V.F. (2019). Metodologias de ensino aplicadas à robótica educacional. *Anais do 14° Simpósio Brasileiro de Automação Inteligente (SBAI 2019)*.

Ecar, M.D.S., Billa, C., and Carvalho, E.L. (2012). Pesquisa em futebol de robôs: 2d simulation league. *Anais do Salão Internacional de Ensino, Pesquisa e Extensão*, 4(2).

Fabro, J.A., Botta, A.L.C., Oenning, B.d.F., and Neri, J.R.F. (2013). O time gpr-2d 2013 de simulação 2d: Avanços no uso de aprendizado por reforço para a tomada de decisão no ataque, e abordagem de modificação de estratégias durante a partida. *Competição Brasileira de Robótica 2013*.

Faria, B.M., Reis, L.P., Lau, N., and Castillo, G. (2010). Machine learning algorithms applied to the classification of robotic soccer formations and opponent teams. In *2010 IEEE Conference on Cybernetics and Intelligent Systems*, 344–349. IEEE.

Garcia, M.F., Ferreira, D., and da Costa, A.L. (2019). Projeto e concepção de uma plataforma para robótica cognitiva embarcada. *Anais do 14° Simpósio Brasileiro de Automação Inteligente (SBAI 2019)*.

Goulart, C., Valadão, C., Caldeira, E.M., and Bastos-Filho, T.F. (2015). Maria: um robô para interação com crianças com autismo. *XII Simpósio Brasileiro de Automação Inteligente (SBAI)*, 557–562.

Guarenti, C.A. and Ribeiro, L.O.M. (2019). Robótica educacional como objeto de aprendizagem: Oficinas para professores em formação. *Anais do Salão Internacional de Ensino, Pesquisa e Extensão*, 10(1).

Henn, T., Henrio, J., and Nakashima, T. (2017). Optimizing player's formations for corner-kick situations in robocup soccer 2d simulation. *Artificial Life and Robotics*, 22(3), 296–300.

Jolly, K., Ravindran, K., Vijayakumar, R., and Kumar, R.S. (2007). Intelligent decision making in multi-agent robot soccer system through compounded artificial neural networks. *Robotics and Autonomous Systems*, 55(7), 589–596.

Júnior, O.V.S., de Souza, J.P.R.P., Linder, M.S., and da Costa, A.L. (2006). Mecateam: Um sistema multiagente para o futebol de robôs simulado baseado no agente autônomo concorrente. *Encontro de Robótica Inteligente / XXVI Congresso da Sociedade Brasileira de Computação*.

Kaelbling, L.P., Littman, M.L., and Moore, A.W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4, 237–285.

Kerbage, S.E., Antunes, E.O., Almeida, D.F., and Rosa, P.F. (2010). Generalização da aprendizagem por reforço: Uma estratégia para robôs autônomos cooperativos. *Competição Latino Americana de Robótica*.

Kitano, H. (1998). *RoboCup-97: robot soccer world cup I*, volume 1395. Springer Science & Business Media.

Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E., and Matsubara, H. (1997). Robocup: A challenge problem for AI. *AI magazine*, 18(1), 73–73.

Li, X. and Chen, X. (2015). Fuzzy inference based forecasting in soccer simulation 2d, the robocup 2015 soccer simulation 2d league champion team. In *Robot Soccer World Cup*, 144–152. Springer.

Martins, M.F. (2007). Aprendizado por reforço acelerado por heurísticas aplicado ao domínio do futebol de robôs. *Programa de Pós-Graduação em Engenharia Elétrica, Centro Universitário da FEI*.

Matarić, M.J. (2014). *Introdução à robótica*. Editora Blucher.

Montgomery, D.C. (2017). *Design and analysis of experiments*. New York: John Wiley & Sons., 9th edition.

Neri, J.R.F., Zatelli, M.R., dos Santos, C.H.F., and Fabro, J.A. (2012). A proposal of qlearning to control the attack of a 2d robot soccer simulation team. In *2012 Brazilian Robotics Symposium and Latin American Robotics Symposium*, 174–178.

Osawa, E., Kitano, H., Asada, M., Kuniyoshi, Y., and Noda, I. (1996). Robocup: The robot world cup initiative. 9–13.

Ottoni, A.L.C., Oliveira, M., Nepomuceno, E.G., and Lamperti, R.D. (2015). Análise do aprendizado por reforço via modelos de regressão logística: Um estudo de caso no futebol de robôs. *Revista Junior de Iniciação Científica em Ciências Exatas e Engenharia*, 1(10), 44–49.

Ottoni, A.L.C., Nepomuceno, E.G., de Oliveira, M.S., and de Oliveira, D.C.R. (2020a). Tuning of reinforcement learning parameters applied to sop using the scott–knott method. *Soft Computing*, 24, 4441–4453.

Ottoni, A.L., Nepomuceno, E.G., and de Oliveira, M.S. (2018). A response surface model approach to parameter estimation of reinforcement learning for the travelling salesman problem. *Journal of Control, Automation and Electrical Systems*, 29(3), 350–359.

Ottoni, A.L.C., Nepomuceno, E.G., and de Oliveira, M.S. (2020b). Development of a pedagogical graphical interface for the reinforcement learning. *IEEE Latin America Transactions*, 18(01), 92–101.

Pêgo, J.M.T., Carvalho, J.C.M., and Gonçalves, R.S. (2015). Estudo de uma nova estrutura robótica paralela combinada. *Blucher Mathematical Proceedings*, 1(1), 159–168.

Pellegrini, J. and Wainer, J. (2007). Processos de decisão de markov: um tutorial. *Revista de Informática Teórica e Aplicada*, 14(2), 133–179.

Pinto, C.A.S. (2019). *A Gestão do Conhecimento e a Inteligência Colaborativa Em Ambientes de Aprendizagem. Um Estudo a Partir da Oficina de Robótica Educacional no Colégio Militar do Rio de Janeiro*. Ph.D. thesis.

Prokopenko, M., Wang, P., Marian, S., Bai, A., Li, X., and Chen, X. (2018). Robocup 2d soccer simulation league: Evaluation challenges. In H. Akiyama, O. Obst, C. Sammut, and F. Tonidandel (eds.), *RoboCup 2017: Robot World Cup XXI*, 325–337. Springer International Publishing, Cham.

Puterman, M.L. (2014). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Romero, R.A., Prestes, E., Osório, F., and Wolf, D. (2014). Robótica móvel. *São Paulo: LTC*.

Schweighofer, N. and Doya, K. (2003). Meta-learning in reinforcement learning. *Neural Networks*, 16(1), 5–9.

Silva, L. and Simoes, M. (2009). Uma estratégia para seleção de agentes heterogênos para o futebol de robôs simulado. *Núcleo de Arquitetura de Computadores e Sistemas Operacionais (ACSO), UNEB*.

Sutton, R.S. and Barto, A.G. (2018). *Reinforcement learning: An introduction*. MIT press.

Watkins, C.J. and Dayan, P. (1992). Technical note Q-learning. *Machine Learning*, 8(3), 279–292.